Trajectory tracking design for linear complementarity systems with continuous solutions

Van Nam Vo

UNIVERSITY OF LIMOGES
FACULTY OF SCIENCE AND TECHNOLOGY

INTERNSHIP REPORT
for the degree of Master in Applied Mathematics

TRAJECTORY TRACKING DESIGN FOR LINEAR COMPLEMENTARITY SYSTEMS WITH CONTINUOUS SOLUTIONS

Author:
Van Nam VO

Supervisors:
Prof. Bernard BROGLIATO
Prof. Christophe PRIEUR

LIMOGES, 2019

Acknowledgements

First and foremost, I wish to express my gratitude to my supervisors, Prof. Bernard BROGLIATO and Prof. Christophe PRIEUR, for their enthusiasm and continuous guidance, helped me to complete my thesis. Their guidance helped me all the time of this internship and in the writing of this report. I am thankful for the chance to learn from them.

I wish to thank my Professors at the University of Limoges, who gave me precious courses during my master training both on-site and online. Also, my special thanks go to my teachers at the University of Quy Nhon, Vietnam for their help and support during the first year of the master degree.

I wish to thank my friends in TRIPOP Team, INRIA Grenoble, my classmates at the University of Limoges and my Vietnamese friends for their friendship, suggestions, encouragement and for making a friendly environment.

Finally, I wish to express my special gratitude to my family. Without their unconditional love and support, I would not be able to do what I have accomplished.

Limoges, June 25th, 2019

Student.

Van Nam VO

Contents

List of Figures

Introduction

With a large range of applications in various fields like mechanics, electrical circuits, control systems, etc., *complementarity systems* have recently been the object of strong attention. In mathematical programming, complementarity conditions are well-known and they arise, for instance, in the Karush–Kuhn–Tucker conditions for optimality. In terms of electrical circuits, imposing complementarity conditions simply means that some ports are terminated by ideal diodes (see [6]).

Because of its applications, like other dynamical systems, dynamic systems with complementarity conditions, called *complementarity dynamical systems*, have been carefully studied as witnessed by many articles, see for instance [2,3,5,6,8,9].

The well-posedness has been treated and another interesting problem has arisen. That is controlling such dynamical systems and it is also our object in this report. Due to some limitations, in this paper, we restricted our attention to linear underlying dynamics with consistent initialization, such that no state jump occurs (solutions are continuous). Our goals are:

- To survey the well-posedness of Linear Complementarity System,
- To find a controller such that the state x and desired trajectory x_d satisfy $\|x(t) - x_d(t)\| \to 0$, as $t \to +\infty$.

This work was developed in my internship funded by INRIA (The French National Institute for Research in Computer Science and Control) from February to July 2019. The report consists of two main chapters, Introduction, Conclusion and Appendix.

Chapter 1 is devoted to the study of the existence and uniqueness of solution of Linear Complementarity Systems with feedthrough matrix satisfying $D = 0$, $D \succeq 0$ and $D \succ 0$, which arise from physical examples, for instance, electrical circuits with a network of ideal diodes. The well-posedness results of the multivalued Lur'e system have been considered as a generalization of so-called Linear Complementarity Systems. That may be important in practice in case D is a nonzero feedthrough matrix, possibly positive semidefinite and non-symmetric.

In **Chapter 2**, our main results are stated and proved to reach our target, that is designing controller u for a Linear Complementarity Systems satisfying reference inputs.

A brief introduction to the SICONOS Platform and modeling of a nonsmooth dynamic system via examples, in the scalar case as well as the higher dimensions, aims at obtaining numerical simulations of these problems. Furthermore, we also deal with extended controller form and a way to choose the "best" gain K.

In **Appendix** we recall some basic facts and notations which will be used throughout this report. It includes some definitions and properties in *Convex analysis* concerning convex sets and convex functions. The solution concept of a dynamic system such as *equilibrium point*, *stability* was introduced as well. Appendix also provides detailed scripts written in C++ and MATLAB that we have used in Chapter 2. We can find there the codes to find feedback gain K and script to execute in SICONOS.

Chapter 1

Well-posedness of Linear Complementarity Systems

In this chapter, we will deal with Linear Complementarity Systems (LCS) and focus on the existence and uniqueness of their solutions, which are absolutely continuous (AC). Three sub-classes of LCS are treated, including $D = 0, D \succeq 0$ and $D \succ 0$ that come from physical examples, specifically electrical circuits with resistors, inductors, capacitors and ideal diodes (RLCD). In order to summarize briefly, the following results are presented without their proofs. The references here are [3,5,6].

1.1 Linear Complementarity Systems

Let us now consider the following complementarity dynamical system

$$\begin{cases} \dot{x}(t) = a(x(t)) + b(x(t))\lambda(t) + e(x(t), u(t)) \\ 0 \leq \lambda(t) \perp y(t) = c(x(t)) + g(u(t)) \geq 0, \end{cases}$$

where $u(t) \in \mathbb{R}^m, y(t) \in \mathbb{R}^p$. This class is very general and needs to be restricted to obtain well-posedness results. Thus, in this paper, we will restrict our attention to the linear complementarity systems of the form

$$\begin{cases} x(0) = x_0, \\ \dot{x}(t) = Ax(t) + B\lambda(t) + Eu(t), \\ y(t) = Cx(t) + D\lambda(t) + Fu(t), \\ 0 \leq \lambda(t) \perp y(t) \geq 0, \end{cases} \tag{1.1}$$

where $A, B, C, D, E,$ and F are real constant matrices with appropriate sizes, $x(t) \in \mathbb{R}^n, \lambda(t) \in \mathbb{R}^p, u(t) \in \mathbb{R}^m$. The last line is a complementarity relation between $\lambda(t)$

and $y(t)$, which are forced to remain non-negative and always orthogonal one to each other. At each time t, both $\lambda_i(t)$ and $y_i(t)$ must be non-negative, and at least one of them should be zero. Using the equivalence

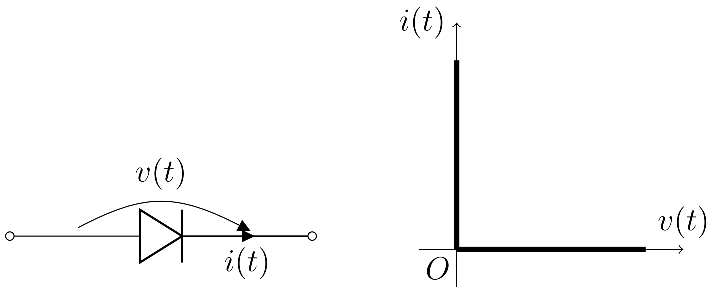
$$0 \leq \lambda(t) \perp y(t) \geq 0 \Leftrightarrow -\lambda(t) \in \partial\psi_{\mathbb{R}^p_+}(y(t)),$$

which holds for vectors $\lambda(t), w(t) \in \mathbb{R}^p$, the LCS above is equivalently rewritten as

$$\begin{cases} x(0) = x_0, \\ \dot{x}(t) = Ax(t) + B\lambda(t) + Eu(t), \\ \lambda(t) \in -\partial\psi_{\mathbb{R}^p_+}(Cx(t) + D\lambda(t) + Fu(t)). \end{cases} \tag{1.2}$$

We define the Linear Complementarity Problem (LCP) with data $q \in \mathbb{R}^m$ and $M \in \mathbb{R}^{m\times m}$ by the problem of finding $z \in \mathbb{R}^m$ such that $0 \leq z \perp q + Mz \geq 0$. We know that this LCP has a unique solution for any q if and only if M is an $m \times m$ P-matrix. A square matrix is said to be a *P-matrix* if all its principal minors are positive (see [10] for more details).

Remark 1.1. Complementarity relations appear in many fileds. For instance, voltage-current characteristic of an ideal diode is a complementarity relation.



(a) Ideal diode (b) Ideal diode characteristic

Figure 1.1: A complementarity relation in physics

Before starting the study of linear complementarity systems with external inputs and outputs, we first need to establish some results on existence and uniqueness of solutions. The following results are taken from [3, 5, 6].

1.1.1 Case $D = 0$

Let us recall some facts and developments from [5].

Assumption 1.1. There exists a constant matrix $P = P^\top \succ 0$ such that

$$PB = C^\top.$$

Denoting by R, the symmetric positive definite square root of P, and letting $z = Rx$, one gets from (1.1)

$$\begin{cases} \dot{z}(t) = R\dot{x}(t) = RAR^{-1}z(t) + REu(t) + RB\lambda(t) \\ 0 \leq \lambda(t) \perp y(t) = CR^{-1}z(t) + Fu(t) \geq 0. \end{cases} \tag{1.3}$$

Since the equivalence

$$0 \leq \lambda(t) \perp CR^{-1}z(t) + Fu(t) \geq 0 \Leftrightarrow -\lambda(t) \in \partial\psi_{\mathbb{R}^p_+}(CR^{-1}z(t) + Fu(t)),$$

consequently, one can rewrite (1.3) as

$$-\dot{z}(t) \in -RAR^{-1}z(t) - REu(t) + RB\partial\psi_{\mathbb{R}^p_+}(CR^{-1}z(t) + Fu(t)).$$

Now using $R^2B = C^\top$, it follows that

$$-\dot{z}(t) \in -RAR^{-1}z(t) - REu(t) + R^{-1}C^\top\partial\psi_{\mathbb{R}^p_+}(CR^{-1}z(t) + Fu(t)). \tag{1.4}$$

For each $t \in [0, +\infty[$ the closed set

$$K(t) := \{x \in \mathbb{R}^n : Cx + Fu(t) \geq 0\}$$

and $\mathbb{R}^m$ are convex polyhedral and $\psi_{K(t)}(x) = (\psi_{\mathbb{R}^m_+ - Fu(t)} \circ C)(x)$. Therefore,

$$C^\top\partial\psi_{\mathbb{R}^p_+}(CR^{-1}z(t) + Fu(t)) = \partial\psi_{K(t)}(x)$$

for any $x \in \mathbb{R}^n$. So the inclusion in (1.4) can be rewritten as

$$-\dot{z}(t) + RAR^{-1}z(t) + REu(t) \in R^{-1}\partial\psi_{K(t)}(R^{-1}z(t)).$$

Consider the closed convex polyhedral set

$$S(t) := R(K(t)) = \{Rx : x \in K(t)\},$$

we can see that $\psi_{S(t)}(x) = (\psi_{K(t)} \circ R^{-1})(x)$ for all $x \in \mathbb{R}^n$. Since R is invertible and symmetric, again we have

$$\partial\psi_{S(t)}(x) = R^{-1}(\partial\psi_{K(t)})(R^{-1}x) \text{ for all } x \in \mathbb{R}^n$$

and hence, since $\mathrm{N}_{S(t)}(x) = \partial\psi_{S(t)}(x)$, the differential inclusion may be written in the form

$$-\dot{z}(t) + RAR^{-1}z(t) + REu(t) \in \mathrm{N}_{S(t)}(z(t)). \tag{1.5}$$

Remark 1.2. By the definition of the normal cone, the inclusion (1.5) is equivalent to the evolution variational inequality

$$\langle \dot{z}(t) - RAR^{-1}z(t) - REu(t), v - z(t)\rangle \geq 0, \forall v \in S(t), z(t) \in S(t).$$

If $F = 0$ then K does not vary with the time and Theorem 2.2 in [8] may applied with $u(\cdot)$ a continuous mapping with locally L^1 derivative. Consequently, we conclude the well-posedness of the LCS. However, here we let K hence S be time-varying, which complicates the analysis.

Theorem 1.1 ([5], Theorem 3.5)**.** *Assume that $u(\cdot) \in L^1_{\text{loc}}(\mathbb{R}_+; \mathbb{R}^p)$ and the set-valued mapping $S(\cdot) = R(K(\cdot))$ is locally absolutely continuous with nonempty values. Then,*

$$-\dot{z}(t) + RAR^{-1}z(t) + REu(t) \in \mathrm{N}_{S(t)}(z(t))$$

with initial condition $z(0) = z_0 \in R(K(0))$ has a unique locally absolutely continuous solution $z(\cdot)$ on $[0, +\infty[$.

Example 1.1. The circuit in Figure 1.2 has the dynamics

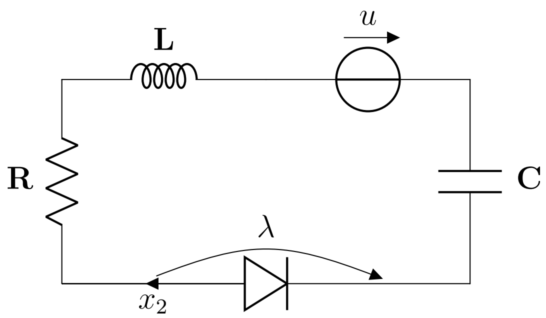


Figure 1.2: RLC circuit with an ideal diode

$$\begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = -\frac{1}{\mathbf{LC}}x_1(t) - \frac{\mathbf{R}}{\mathbf{L}}x_2(t) - \frac{1}{\mathbf{L}}\lambda(t) + \frac{1}{\mathbf{L}}u(t) \\ 0 \leq \lambda(t) \perp -x_2(t) \geq 0 \end{cases}$$

where u denotes the voltage applied to the system, for instance $u(t) = \sin(t)$, $t \geq 0$, x_1 is the charge of the capacitor and x_2 is current though inductor. The matrices A, B, C, D, E, F can be easily identified. In this situation, we have $D = 0$. Following the same lines, we can rewrite the dynamics as

$$-\dot{z}(t) + \begin{pmatrix} 0 & 1 \\ -\frac{1}{\mathbf{LC}} & -\frac{\mathbf{R}}{\mathbf{L}} \end{pmatrix} z(t) + \begin{pmatrix} 0 \\ -\frac{1}{\mathbf{L}\sqrt{\mathbf{L}}} \end{pmatrix} u(t) \in \mathrm{N}_S(z(t)).$$

Here $z = Rx$ with $R = \frac{1}{\sqrt{\mathbf{L}}} I$ and $S = \{(x_1, x_2) \in \mathbb{R}^2 : x_2 \leq 0\}$.

Since $F = 0$, $S(\cdot) = S$ is locally absolutely continuous (see Proposition 3.2, [5]). Now, applying Theorem 1.1, we claim that for any $z_0 \in S$, the dynamics with initial condition $z(0) = z_0$ has a unique locally absolutely continuous solution $z(\cdot)$ on $[0, +\infty[$.

1.1.2 Case $D \succeq 0$

In this section our goal is just to point out that extensions of the above developments led for $D = 0$ are possible when $D \neq 0$. We first consider the positive semidefinite matrix D of the form

$$D = \begin{pmatrix} D_1 & 0 \\ 0 & 0 \end{pmatrix},$$

where $D_1 \succ 0$ is square of dimension $q < p$, and not necessarily symmetric. We partition the multiplier as $\lambda = \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix}$, such that $D\lambda = \begin{pmatrix} D_1\lambda_1 \\ 0 \end{pmatrix}$. The vector y is partitioned similarly. Then the complementarity conditions in (1.1) can be written as the following

$$\begin{cases} 0 \leq \lambda_1(t) \perp y_1(t) = C_1 x(t) + D_1\lambda_1(t) + F_1 u(t) \geq 0, \\ 0 \leq \lambda_2(t) \perp y_2(t) = C_2 x(t) + F_2 u(t) \geq 0. \end{cases}$$

The first set of conditions is a LCP with unknown λ_1 and matrix D_1. It possesses a unique solution that is continuous piecewise linear in $x(t)$ and $u(t)$, and we denote as $\lambda_1(x, u)$. The LCS in (1.1) is rewritten as

$$\begin{cases} \dot{x}(t) = Ax(t) + B_1\lambda_1(x, u) + B_2\lambda_2(t) + Eu(t) \\ 0 \leq \lambda_2(t) \perp y_2(t) = C_2 x(t) + F_2 u(t) \geq 0. \end{cases}$$

If we assume that $P_2 B_2 = C_2^\top$ for some symmetric positive definite matrix P_2, and R_2 the symmetric positive definite square root of P_2, then following the same steps as in case $D = 0$, one gets

$$-\dot{z}(t) + R_2(AR_2^{-1}z(t) + B_1\lambda_1(R_2^{-1}z(t), u(t))) + R_2 Eu(t) \in \mathrm{N}_{S(t)}(z(t)),$$

here $S(t) = R_2(K(t)) = \{R_2 x : x \in K(t)\}$ and $K(t) = \{x \in \mathbb{R}^n : C_2 x + F_2 u(t) \geq 0\}$.

Then Theorem 1.1 may be applied since the steps of its proof can be redone for a Lipschitz continuous field.

Example 1.2. Let us consider the electrical system (Figure 1.3) taken from [3] that is composed of two resistors $\mathbf{R}$ and four capacitors $\mathbf{C}$ and two ideal diodes with characteristics $0 \leq v_1(t) \perp i_1(t) \geq 0$ and $0 \leq v_2(t) \perp i_3(t) \geq 0$, respectively. The state variables are $x_1(t) = \int_0^t i_1(t)\mathrm{d}t, x_2(t) = \int_0^t i_2(t)\mathrm{d}t, x_3(t) = v_2(t)$, and $\lambda_1(t) = i_3(t), \lambda_2(t) = v_1(t)$.

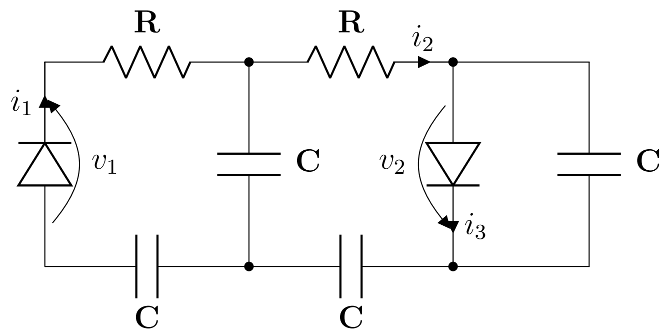


Figure 1.3: Electrical circuit with capacitors, resistors and ideal diodes.

The dynamics is given by

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{pmatrix} = \begin{pmatrix} \frac{-2}{\mathbf{RC}} & \frac{1}{\mathbf{RC}} & 0 \\ \frac{1}{\mathbf{RC}} & \frac{-2}{\mathbf{RC}} & \frac{1}{\mathbf{R}} \\ \frac{-1}{\mathbf{RC}^2} & \frac{2}{\mathbf{RC}^2} & \frac{-1}{\mathbf{RC}} \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} + \begin{pmatrix} 0 & \frac{1}{\mathbf{R}} \\ 0 & 0 \\ \frac{1}{\mathbf{C}} & 0 \end{pmatrix} \begin{pmatrix} \lambda_1(t) \\ \lambda_2(t) \end{pmatrix},$$

$$0 \leq \lambda(t) \perp y(t) = \begin{pmatrix} 0 & 0 & 1 \\ \frac{-2}{\mathbf{RC}} & \frac{1}{\mathbf{RC}} & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{\mathbf{R}} \end{pmatrix} \lambda(t) \geq 0.$$

The matrices A, B, C and D are easily identified. Set $B_1 = (0, 0, \frac{1}{\mathbf{C}})^\top$, $B_2 = (\frac{1}{\mathbf{R}}, 0, 0)^\top$, $C_1 = (0, 0, 1)$, and $C_2 = (\frac{-2}{\mathbf{RC}}, \frac{1}{\mathbf{RC}}, 0)$. The linear complementarity conditions can be rewritten as follows

$$\begin{cases} 0 \leq \lambda_1(t) \perp C_1 x(t) \geq 0, \\ 0 \leq \lambda_2(t) \perp C_2 x(t) + \dfrac{1}{\mathbf{R}} \lambda_2(t) \geq 0. \end{cases} \tag{1.6}$$

From the second line, we have $\lambda_2(t) = -\mathbf{R} \min\{0, C_2 x(t)\}$. The matrix D is positive semidefinite and relation $P_1 B_1 = C_1^\top$ holds with $P_1 = \mathbf{C} I_3$. Let $z = R_1 x$ with $R_1 = \sqrt{\mathbf{C}} I_3$, one obtains the differential inclusion

$$-\dot{z}(t) + Az(t) - \frac{\mathbf{R}}{\sqrt{\mathbf{C}}} B_2 \min\{0, C_2 z(t)\} \in \mathrm{N}_{S(t)}(z(t)),$$

where $K(t) = \{x \in \mathbb{R}^3 : C_1 x \geq 0\} = \{(x_1, x_2, x_3) \in \mathbb{R}^3 : x_3 \geq 0\}$ and $S(t) = \{R_1 x : x \in K(t)\} = \{\sqrt{\mathbf{C}}(x_1, x_2, x_3) \in \mathbb{R}^3 : x_3 \geq 0\}$. Since the steps of the proof of Theorem 1.1 can be redone for a Lipschitz continuous field, we can conclude that the dynamics has a unique solution with an initial condition $x_0 = (x_1, x_2, x_3)$ satisfying $x_3 \geq 0$.

Now we will deal with the matrix $D \succeq 0$ but not in the form $D = \begin{pmatrix} D_1 & 0 \\ 0 & 0 \end{pmatrix}$. However, the following lemma gives us a way to feedthrough matrix D.

Lemma 1.1 ([5], Lemma 3.8)**.** *Let $D \in \mathbb{R}^{p\times p}$ have rank $q < p$, and suppose that there exist full-rank matrices V and W with $W^\top = V$, such that $D = V\bar{D}W$ with $\bar{D} = \begin{pmatrix} D_1 & 0 \\ 0 & 0 \end{pmatrix}$, where D_1 is a $q \times q$ full-rank matrix. Then the LCS in (1.1) can be equivalently rewritten as the cone complementarity system (CCS)*

$$\begin{cases} \dot{x}(t) = Ax(t) + Eu(t) + BW^{-1}\bar{\lambda}(t) \\ C_{W^{-1}} \ni \bar{\lambda}(t) \perp \bar{y}(t) = V^{-1}Cx(t) + V^{-1}Fu(t) + \bar{D}\bar{\lambda}(t) \in C_V \end{cases} \tag{1.7}$$

where $C_V = \{z \in \mathbb{R}^p : Vz \in \mathbb{R}^p_+\}$, and $C_{W^{-1}} = \{z \in \mathbb{R}^p : W^{-1}z \in \mathbb{R}^p_+\}$ are two dual polyhedral cones.

The second line in (1.7) is so-called a cone complementarity problem (CCP).

Remark 1.3. Following the same lines as in the case $D = 0$ and in the case matrix D of the form $\begin{pmatrix} D_1 & 0 \\ 0 & 0 \end{pmatrix}$ and since a CPP with two dual cones can be equivalently written as an inclusion into a normal cone, the CCS in (1.7) can be transformed into a sweeping process. Further conditions on D_1 are necessary to ensure that these manipulations are doable.

1.1.3 Case $D \succ 0$

The case D is positive definite is worth stating separately.

Proposition 1.1 ([6], Corollary 10.6)**.** *Let us consider an LCS given by (1.1) such that $D \succ 0$. Then the LCS has a unique continuous solution for any initial state x_0 and input $u(\cdot) \in L^1_{\text{loc}}(\mathbb{R}_+; \mathbb{R}^p)$.*

Remark 1.4. The continuity of the solution is also stated by Corollary 10.6 in [6].

Example 1.3. Let us consider the circuit in Figure 1.4 with a voltage source u. The dynamics is given by

$$\begin{cases} \dot{x_1}(t) = -\frac{1}{\mathbf{RC}}x_1(t) + x_2(t) - \frac{\lambda(t)}{\mathbf{R}} + \frac{u(t)}{\mathbf{R}} \\ \dot{x_2}(t) = -\frac{1}{\mathbf{LC}}x_1(t) - \frac{\lambda(t)}{\mathbf{L}} + \frac{u(t)}{\mathbf{L}} \\ 0 \le \lambda(t) \perp \frac{\lambda(t)}{\mathbf{R}} + \frac{x_1(t)}{\mathbf{RC}} - x_2(t) - \frac{u(t)}{\mathbf{R}} \ge 0. \end{cases} \tag{1.8}$$

Here, we considered the current through the inductor for the variable x_2, and for the variable x_1 the charge on the capacitor as state variables. It can be seen that (1.8) is an

LCS with matrix $D = \frac{1}{\mathbf{R}} > 0$. According to Proposition 1.1, we claim that this problem has a unique continuous solution for any initial state x_0 and input u.

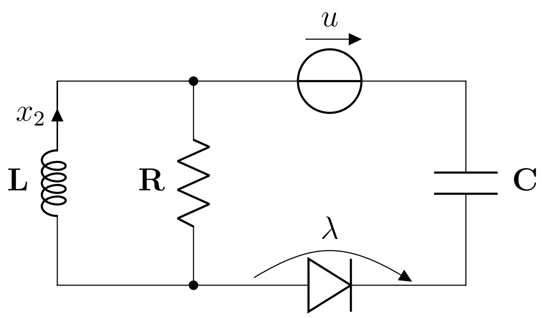


Figure 1.4: A circuit with an ideal diode in case $D \succ 0$

In our framework, the assumption on existence a matrix P that is symmetric, positive definite satisfying $PB = C^\top$ is significant to perform the steps after; however, in a certain situation, finding a such matrix is not easy even difficult. The following proposition provides a sufficient condition for existence of matrix P. It is considered as a corollary of Lemma 10.4 in [6].

Proposition 1.2. *Let $B \in \mathbb{R}^{n\times p}, C \in \mathbb{R}^{p\times n} \in$ and let B be full column-rank. Then, there exists a symmetric positive definite matrix P such that $PB = C^\top$ if and only if CB is symmetric positive definite.*

1.2 The multivalued Lur'e system

Suppose that $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times p}, C \in \mathbb{R}^{p\times n}, D \in \mathbb{R}^{p\times p}$ are given matrices, $f \in C^0(\mathbb{R}_+;\mathbb{R})$ such that $\dot{f} \in L^1_{\text{loc}}(\mathbb{R}_+;\mathbb{R}^n)$ and $\varphi_i : \mathbb{R} \to \mathbb{R} \cup \{+\infty\}(1 \le i \le p)$ given proper convex and lower semicontinuous functions. Let $x_0 \in \mathbb{R}^n$ be some initial condition, we consider the problem: Find $x \in C^0(\mathbb{R}_+;\mathbb{R}^n)$ such that $\dot{x} \in L^\infty_{\text{loc}}(\mathbb{R}_+;\mathbb{R}^n)$ and x right-differentiable on $\mathbb{R}_+$, $\lambda \in C^0(\mathbb{R}_+;\mathbb{R}^p)$ and $y \in C^0(\mathbb{R}_+;\mathbb{R}^p)$ satisfying the nonsmooth dynamical system

$$\begin{cases} x(0) = x_0 \\ \dot{x}(t) = Ax(t) + B\lambda(t) + f(t) \text{ a.e. } t \ge 0 \\ y(t) = Cx(t) + D\lambda(t) \\ \lambda_1(t) \in -\partial\varphi_1(y_1(t)) \\ \lambda_2(t) \in -\partial\varphi_2(y_2(t)) \\ \qquad \vdots \\ \lambda_p(t) \in -\partial\varphi_p(y_p(t)) \end{cases} \tag{1.9}$$

The case $D = 0$ was considered in [5], so we shall study the case $D \neq 0$. The following results can be found in [3]. Let us now denote by p_I (and set $p_{II} = p - p_I$) the largest integer such that the matrix D can be written as follows:

$$D = \begin{pmatrix} 0_{p_I \times p_I} & 0_{p_I \times p_{II}} \\ 0_{p_{II} \times p_I} & D_{II} \end{pmatrix}$$

with $D_{II} \neq 0_{p_{II} \times p_{II}}$. In using this notation, it is convenient to suppose that $p_I = 0$ (resp. $p_{II} = 0$) means that the terms indexed by I (resp. II) are useless and not considered. For $z \in \mathbb{R}^{p_I}$, we set $z = (z_I \quad z_{II})^\top$ with $z_I \in \mathbb{R}^{p_I}$ and $z_{II} \in \mathbb{R}^{p_{II}}$,

$$B = \begin{pmatrix} B^I & B^{II} \end{pmatrix}, C = \begin{pmatrix} C_I \\ C_{II} \end{pmatrix},$$

here $B^I \in \mathbb{R}^{n \times p_I}, B^{II} \in \mathbb{R}^{n \times p_{II}}, C_I \in \mathbb{R}^{p_I \times n}$ and $C_{II} \in \mathbb{R}^{p_{II} \times n}$. Finally, we set

$$\forall y \in \mathbb{R}^{p_I} : \Phi_I(y) := \varphi_1(y_1) + \varphi_2(y_2) + \cdots + \varphi_{p_I}(y_{p_I})$$

and

$$\forall y \in \mathbb{R}^{p_{II}} : \Phi_{II}(y) := \varphi_{p_I+1}(y_1) + \varphi_2(y_{p_I+2}) + \cdots + \varphi_p(y_{p_{II}}).$$

We also set for all $y \in \mathbb{R}^p$, and $z \in \mathbb{R}$

$$\Phi(y) = \Phi_I(y_I) + \Phi_{II}(y_{II}) \text{ and } \Phi^{*,-}(z) = \Phi^*(-z).$$

By using these notations, we can see that the nonsmooth dynamical system reduces to the system

$$\begin{cases} x(0) = x_0 \\ \dot{x}(t) = Ax(t) + B^I\lambda_I(t) + B^{II}\lambda_{II}(t) + f(t) \text{ for all } t \geq 0 \\ y_I(t) = C_I x(t) \\ \lambda_I(t) \in -\partial\Phi_I(y_I(t)) \\ y_{II}(t) = C_{II}x(t) + D_{II}\lambda_{II}(t) \\ y_{II}(t) \in -\partial\Phi_{II}^{*,-}(\lambda_{II}(t)). \end{cases} \tag{1.10}$$

Assumption 1.2. For each $1 \leq i \leq p$, there exists $z_{0,i} \in \mathbb{R}$ at which $\varphi_i^{*,-}(\cdot)$ is continuous, with

$$\varphi_i^{*,-}(z) = \varphi_i^*(-z), \forall z \in \mathbb{R}.$$

Assumption 1.2 is required to guarantee that for all $z \in \mathbb{R}$,

$$\partial \varphi_i^{*,-}(z) = -\partial \varphi_i^*(-z).$$

Then

$$\lambda_i \in -\partial \varphi_i(y_i) \Leftrightarrow y_i \in -\partial \varphi_i^{*,-}(\lambda_i).$$

Assumption 1.3 (If $p_I \geq 1$)**.** There exists a symmetric and invertible matrix $W \in \mathbb{R}^{n \times n}$ such that

$$W^2 B^I = C_I^\top.$$

Then, we set

$$V = \begin{cases} W & \text{if } p_I \geq 1 \\ I & \text{if } p_I = 0 \end{cases}$$

and

$$\forall w \in \mathbb{R}^{p_I}, \Xi_I(w) = \begin{cases} \Phi_I(C_I V^{-1} w) & \text{if } p_I \geq 1 \\ 0 & \text{if } p_I = 0. \end{cases}$$

Assumption 1.4 (If $p_I \geq 1$)**.** There exists a point w_0 in $\mathbb{R}^{p_I}$ at which $\Xi_I(\cdot)$ is continuous.

Assumptions 1.3 and 1.4 ensure that in case $p_I \geq 1$, one has for all $w \in \mathbb{R}^{p_I}$,

$$\partial \Xi_I(w) = V^{-T} C_I^\top \partial \Phi_I(C_I V^{-1} w) = V^{-1} C_I^\top \partial \Phi_I(C_I V^{-1} w).$$

The multivalued mapping $\Xi_I(\cdot)$ is maximal monotone, being the subdifferential of a convex, proper, lower semicontinuous function. Let us denote

$$\forall x \in \mathbb{R}^n, \Lambda_{II}(x) = \begin{cases} V B^{II}(D_{II} + \partial \Phi_{II}^{*,-})^{-1}(-C_{II} V^{-1} x) & \text{if } p_{II} \geq 1 \\ 0 & \text{if } p_{II} = 0. \end{cases}$$

Assumption 1.5 (If $p_{II} \geq 1$)**.** The operator $\Lambda_{II} : \mathbb{R}^n \to \mathbb{R}^n, x \mapsto \Lambda_{II}(x)$ is well-defined, single-valued and Lipschitz continuous.

Proposition 1.3 ([3], Proposition 1)**.** *Suppose that D_{II} is positive definite and $\Phi_{II}(\cdot)$ is proper convex and lower semicontinuous. Then the operator $(D_{II} + \partial \Phi_{II}^{*,-})^{-1}$ is well-defined, single-valued and Lipschitz continuous.*

Proposition 1.4 ([3], Proposition 2)**.** *Suppose that D_{II} is a P-matrix and $\Phi_{II}(\cdot) = \psi_{\mathbb{R}_+^{p_{II}}}(\cdot)$. Then the operator $(D_{II} + \partial \Phi_{II}^{*,-})^{-1}$ is well-defined, single-valued and Lipschitz continuous.*

We recall that for a nonempty closed and convex set $C \subset \mathbb{R}^n$, the recession cone of C is defined by

$$C_\infty = \bigcap_{\lambda > 0} \frac{1}{\lambda}(C - x_0),$$

where x_0 is an element in C. We also denote by $\ker(M)$ the kernel of M and by $\mathcal{R}(M)$ the range of M.

Proposition 1.5 ([3], Proposition 3)**.** *Suppose that D_{II} is positive semidefinite,* $\mathrm{dom}(\Phi_{II}^{*,-}) = \overline{\mathrm{dom}(\Phi_{II}^{*,-})}$,

$$(\mathrm{dom}(\Phi_{II}^{*,-}))_\infty \cap \ker(D_{II} + D_{II}^\top) \cap \{z \in \mathbb{R}^{p_{II}} : D_{II} z \in ((\mathrm{dom}(\Phi_{II}^{*,-}))_\infty)^*\} = \{0\}.$$

Moreover,

$$\mathcal{R}(C_{II}) \subset \mathcal{R}(D_{II} + D_{II}^\top) \subset \ker(B^{II}).$$

Then the operator $x \mapsto VB^{II}(D_{II} + \partial\Phi_{II}^{*,-})^{-1}(-C_{II}V^{-1}x)$ *is well-defined, single-valued and Lipschitz continuous.*

The problem in the nonsmooth dynamical system (1.10) can be reduced, by setting $X(t) = Vx(t)$, $\forall t \geq 0$, to the following dynamical variational inequality problem: Find $X \in C^0(\mathbb{R}_+; \mathbb{R}^n)$ such that $\dot{X} \in L^\infty_{\mathrm{loc}}(\mathbb{R}_+; \mathbb{R}^n)$ and X right-differentiable on $\mathbb{R}_+$ with $X(0) = X_0$ and satisfying for almost everywhere $t \geq 0$ the variational inequality

$$\langle \dot{X}(t) - VAV^{-1}X(t) - \Lambda_{II}(X(t) - Vf(t), v - X(t)\rangle + \Xi_I(v) - \Xi_I(X(t)) \geq 0,$$

holds for all $v \in \mathbb{R}^n$.

Indeed, in case $p_I \geq 1$ and $p_{II} \geq 1$, from (1.10), it is clear that

$$\dot{x}(t) - B^{II}(D_{II} + \partial\Phi_{II}^{*,-})^{-1}(-C_{II}x(t)) - Ax(t) - f(t) \in -B^I \partial\Phi_I(C_I x(t)).$$

Notice that

$$D_{II}z - q \in -\partial\Phi_{II}^{*,-}(z) \Leftrightarrow z \in (D_{II} + \partial\Phi_{II}^{*,-})^{-1}(q),$$

for all $q \in \mathbb{R}^{p_I}$.
Multiply by invertible matrix V, then we obtain

$$\dot{X}(t) - VAV^{-1}X(t) - VB^{II}(D_{II} + \partial\Phi_{II}^{*,-})^{-1}(-C_{II}V^{-1}X(t)) - Vf(t) \in -\partial\Xi_I(X(t))$$

from which one deduces the variational inequality. The case $p_I = 0$ (resp. $p_{II} = 0$) can be deduced from the previous relations by removing the terms indexed by I (resp. II).

1.2.1 Well-posedness by Kato's Theorem

The following result relies on Kato's theorem.

Theorem 1.2 ([3], Theorem 1)**.** *Suppose that Assumptions 1.2-1.5 hold. Then for any $X_0 \in \mathrm{Dom}(\partial\Xi_I)$ there exists a unique $X \in C^0(\mathbb{R}_+;\mathbb{R}^n)$ such that $\dot{X} \in L^\infty_{\mathrm{loc}}(\mathbb{R}_+;\mathbb{R}^n)$, X right-differentiable on $\mathbb{R}_+$ and for all $v \in \mathbb{R}^n$,*

$$\begin{aligned} X(t) \in \mathrm{Dom}(\partial\Xi_I), \forall t \geq 0, \\ \langle \dot{X}(t) - VAV^{-1}X(t) - \Lambda_{II}(X(t) - Vf(t), v - X(t)\rangle + \Xi_I(v) - \Xi_I(X(t)) \geq 0, \\ X(0) = X_0. \end{aligned}$$

We can check that Assumptions 1.2-1.5 are satisfied for the system given in Example 2.2. Then, using Theorem 1.2, one concludes that the dynamical system has a unique solution.

Corollary 1.1. Let $p_I = p$. Suppose that Assumptions 1.2-1.4 hold. Then for any $X_0 \in \mathrm{dom}(\Xi)$ there exists a unique $X \in C^0(\mathbb{R}_+;\mathbb{R}^n)$ such that $\dot{X} \in L^\infty_{\mathrm{loc}}(\mathbb{R}_+;\mathbb{R}^n)$, X right-differentiable on $\mathbb{R}_+$ satisfying

$$\begin{cases} X(t) \in \mathrm{Dom}(\partial\Xi), \forall t \geq 0, \\ \langle \dot{X}(t) - VAV^{-1}X(t) - Vf(t), v - X(t)\rangle + \Xi(v) - \Xi(X(t)) \geq 0, \forall v \in \mathbb{R}^n, \text{a.e. } t \geq 0, \\ X(0) = X_0. \end{cases}$$

It follows that for $x_0 \in \mathbb{R}^n$ such that $Cx_0 \in \mathrm{Dom}(\partial\Phi)$, the function $x(\cdot) = V^{-1}X(\cdot)$ is the unique solution of the problem

$$\begin{cases} x(0) = x_0 \\ \dot{x}(t) = Ax(t) + B\lambda(t) + f(t) \text{ a.e. } t \geq 0 \\ y(t) = Cx(t), \forall t \geq 0 \\ \lambda(t) \in -\partial\Phi(y(t)), \forall t \geq 0 \end{cases}$$

Corollary 1.2. Let $p_{II} = p$. Suppose that Assumptions 1.2 and 1.5 hold. Then for any $X_0 \in \mathbb{R}^n$ there exists a unique $X \in C^0(\mathbb{R}_+;\mathbb{R}^n)$ such that $\dot{X} \in L^\infty_{\mathrm{loc}}(\mathbb{R}_+;\mathbb{R}^n)$, X right-differentiable on $\mathbb{R}_+$ satisfying

$$\begin{cases} \dot{X}(t) - AX(t) - B(D + \partial\Phi^{*,-})^{-1}(-CX(t)) - f(t) = 0, a.e.\, t \geq 0, \\ X(0) = X_0. \end{cases}$$

Then for $x_0 \in \mathbb{R}$ the function $X(\cdot)$ is the unique solution of Problem 1.9, that is

$$\begin{cases} x(0) = x_0 \\ \dot{x}(t) = Ax(t) + B\lambda(t) + f(t) \text{ a.e. } t \geq 0 \\ y(t) = Cx(t) + D\lambda(t), \forall t \geq 0 \\ \lambda(t) \in -\partial\Phi(y(t)), \forall t \geq 0 \end{cases}$$

1.2.2 Well-posedness by maximal monotonicity

It is noteworthy that the condition $\mathcal{R}(C_{II}) \subset \mathcal{R}(D_{II} + D_{II}^\top) \subset \ker(B^{II})$ in Proposition 1.5 implies that the symmetric part of D has a large enough range, but in some cases of D, that is not satisfied. This motivates us to look for another assumptions to show the well-posedness of Problem (1.9).

Assumption 1.6. There exists a symmetric and positive definite matrix P such that $PB = C^\top$ and D is positive semidefinite.

It is clear that the Lur'e system in (1.9) may be rewritten by

$$\dot{x}(t) \in Ax(t) + B(D + \partial\Phi^{*,-})^{-1}(-Cx(t)) + f(t). \tag{1.11}$$

Defining R as $R^2 = P$, the symmetric positive definite square root of P, and letting $z = -Rx$, one gets from (1.11)

$$\dot{z}(t) \in RAR^{-1}z(t) - R^{-1}C^\top(D + \partial\Phi^{*,-})^{-1}(-CR^{-1}z(t)) - Rf(t).$$

Since R is symmetric, the operator $z \mapsto R^{-1}C^\top(D + \partial\Phi^{*,-})^{-1}(CR^{-1}z)$ is maximal monotone, provided $\mathcal{R}(CR^{-1}) \cap \operatorname{rint}(\operatorname{Dom}((D + \partial\Phi^{*,-})^{-1})) \neq \emptyset$.

Theorem 1.3 ([3], Theorem 2)**.** *Let Assumptions 1.2 and 1.6 hold and suppose that* $\mathcal{R}(CR^{-1}) \cap \operatorname{rint}(\operatorname{Dom}((D + \partial\Phi^{*,-})^{-1})) \neq \emptyset$*. Let $x_0 \in \mathbb{R}^n$ such that $-Cx_0 \in$* $\operatorname{Dom}((D + \partial\Phi^{*,-})^{-1})$*. Then the Lur'e system in (1.9) possesses a unique solution that is Lipschitz continuous.*

Before ending this section, we consider the example as follows.

Example 1.4. Let us consider the Filtered full wave rectifier in Figure 1.5 which is taken from [3] with the dynamics given by

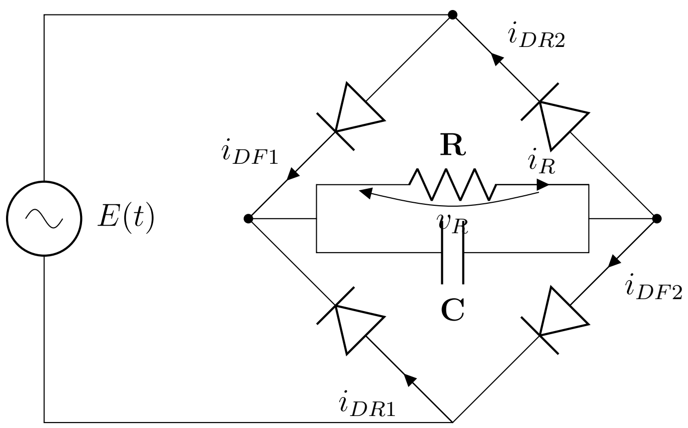


Figure 1.5: Filtered full wave rectifier

$$\begin{cases} \dot{x}(t) = -\frac{1}{\mathbf{RC}}x(t) + \begin{pmatrix} \frac{1}{\mathbf{C}} & 0 & \frac{1}{\mathbf{C}} & 0 \end{pmatrix} \lambda(t) \\ 0 \leq \lambda(t) \perp y(t) = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 1 & -1 \\ 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \lambda(t) + \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} x(t) \geq 0, \end{cases} \tag{1.12}$$

where the state x is the voltage across the capacitor, $y_1 = v_{DR1}, y_2 = i_{DF2}, y_3 = v_{DF1}$, $y_4 = v_{DR2}$, and $\lambda = (i_{DR1}, v_{DF2}, i_{DF1}, i_{DR2})^\top$. The matrix D is positive semidefinite since it is skew symmetric. The relation $PB = C^\top$ holds with $P = \mathbf{C} > 0$. We choose $R = \sqrt{\mathbf{C}}$. Then, $\mathcal{R}(CR^{-1}) = \{\alpha(1,0,1,0)^\top, \alpha \in \mathbb{R}\}$. It is noteworthy that $\mathrm{Dom}((D + \partial\Phi^{*,-})^{-1}) = \mathcal{R}(D + \partial\Phi^{*,-})$ and the condition of Theorem 1.3 is satisfied and the system (1.12) has a unique Lipschitz solution (see [3]).

Remark 1.5. [3] In the case $y = Cx + D\lambda + Fu$ and $f(t) = Eu(t)$, where $u(\cdot)$ is some $m-$dimensional control input, then the results above may be used to analyse feedback controllers of the form $u = Sx + G\lambda$. It suffices to replace (A, B, C, D) by $(A + ES, B + EG, C + FS, D + FG)$ in the analysis.

1.2.3 Dissipativity and stability results

Suppose that the Lur'e system is well-posed, we are now at the position to study stability properties of the system. Denote

$$\mathcal{X}_0 = \begin{cases} \{x \in \mathbb{R}^n : C_I x \in \mathrm{Dom}(\partial\Phi_I)\} & \text{if } p_I \geq 1 \\ \mathbb{R}^n & \text{if } p_I = 0. \end{cases}$$

Then, for $x_0 \in \mathcal{X}_0$, the following problem

$$\begin{cases} \dot{x}(t) = Ax(t) + B\lambda(t) \text{ a.e. } t \geq 0 \\ y(t) = Cx(t) + D\lambda(t), \forall t \geq 0 \\ y(t) \in -\partial\Phi^{*,-}(\lambda(t)), \forall t \geq 0 \\ x(0) = x_0, \end{cases}$$

shortly denoted as the system $(A, B, C, D, \Phi^{*,-})$, has a unique solution

$$\begin{aligned} x(\cdot; x_0) : \mathbb{R}_+ &\to \mathbb{R}^n \\ t &\mapsto \mathbb{R}^n. \end{aligned}$$

In order to make sure that $x(t; 0) = 0$ for all $t \geq 0$, we need one more assumption.

Assumption 1.7. Suppose that the initial data satisfies

$$0 \in \mathcal{X}_0 \text{ and } \lambda_{II}(0) \in \partial\Xi_I(0).$$

The notion of passivity (dissipativity) has played an important role in various contexts such as stability issues, adaptive control, identification, etc. First, let us present some definitions of dissipative systems.

Definition 1.1. The system (A, B, C, D) is called *passive* or *dissipative*, if there exists a symmetric and positive definite matrix $P \in \mathbb{R}^{n\times n}$ such that the matrix

$$\begin{pmatrix} A^\top P + PA & PB - C^\top \\ B^\top P - C & -(D + D^\top) \end{pmatrix} \preceq 0.$$

Definition 1.2. The system (A, B, C, D) is said to be *strictly passive*, if there exists a symmetric and positive definite matrix $P \in \mathbb{R}^{n\times n}$ such that the matrix

$$\begin{pmatrix} A^\top P + PA + \varepsilon P & PB - C^\top \\ B^\top P - C & -(D + D^\top) \end{pmatrix} \preceq 0,$$

for some $\varepsilon > 0$.

Definition 1.3. One says that the system $(A, B, C, D, \Phi^{*,-})$ is *passive* provided that there exists a symmetric and positive definite matrix $P \in \mathbb{R}^{n\times n}$ such that

$$\langle PAx, x\rangle + \langle (PB - C^\top)z, x\rangle - \langle Dz, z\rangle + \Phi^{*,-}(0) - \Phi^{*,-}(z) \leq 0,$$

holds for all $x \in \mathbb{R}^n, z \in \mathbb{R}^p$.

Definition 1.4. One says that the system $(A, B, C, D, \Phi^{*,-})$ is *strictly passive* provided that there exists a symmetric, positive definite matrix $P \in \mathbb{R}^{n\times n}$ and a real $\varepsilon > 0$ such that

$$\langle PAx + \tfrac{1}{2}\varepsilon Px, x\rangle + \langle (PB - C^\top)z, x\rangle - \langle Dz, z\rangle + \Phi^{*,-}(0) - \Phi^{*,-}(z) \leq 0,$$

holds for all $x \in \mathbb{R}^n$, $z \in \mathbb{R}^p$.

Proposition 1.6 ([3], Propositions 4-5)**.** *If the system (A, B, C, D) is passive (strictly passive) and*

$$\forall z \in \mathbb{R}^p, \quad \Phi^{*,-}(z) \geq \Phi^{*,-}(0)$$

then the system $(A, B, C, D, \Phi^{*,-})$ is passive (strictly passive), respectively.

Theorem 1.4 ([3], Theorem 3)**.** *Assume that the system $(A, B, C, D, \Phi^{*,-})$ is passive. Then there exists a constant $c > 0$ such that for each $x_0 \in \mathcal{X}_0$,*

$$\|x(t; x_0)\| \leq c\|x_0\|, \quad \forall t \geq 0.$$

Furthermore, if the system $(A, B, C, D, \Phi^{*,-})$ is strictly passive then the system is globally exponentially stable, i.e.,

$$\|x(t; x_0)\| \leq c\|x_0\|e^{-\alpha t}, \quad \forall t \geq 0,$$

for some $c > 0, \alpha > 0$.

Remark 1.6. The system $(A, B, C, D, \Phi^{*,-})$ is passive that ensures the stability of the trivial solution. However, if the system is strictly passive, then the trivial solution is stable and global attractive as well.

Chapter 2

Controller design for LCS

In this chapter, we will continue to study the LCS given by

$$\begin{cases} \dot{x}(t) = Ax(t) + B\lambda(t) + Eu(t), \\ 0 \leq \lambda(t) \perp Cx(t) + D\lambda(t) + Fu(t) \geq 0, \\ x(0) = x_0, \end{cases} \tag{2.1}$$

where $x(t) \in \mathbb{R}^n, u(t) \in \mathbb{R}^m, \lambda(t) \in \mathbb{R}^p$ with $D = 0, D \succeq 0$, and $D \succ 0$. As pointed out in Chapter 1, it is worth considering such cases because many systems like electrical circuits leading us to study LCS. In case $D \succeq 0$, we restrict to positive semidefinite matrices D of the form $\begin{pmatrix} D_1 & 0 \\ 0 & 0 \end{pmatrix}$, where $D_1 \succ 0$ is square of dimension $q < p$.

Our goal is to design the controller u such that $\|x(t) - x_d(t)\| \to 0$ as $t \to \infty$, where x_d is a desired function. The main idea comes from the conclusions in Chapter 1. Generally, to find a controller satisfying a given reference is a hard problem. Thus our ambitious in this paper will only deal with some sub-classes of problems which we can handle.

The main references here are [1, 3, 4, 6, 9, 10].

2.1 Sufficient conditions for stability

Recall that the quadruple (A, B, C, D) is called passive if

$$\begin{pmatrix} A^\top P + PA & PB - C^\top \\ B^\top P - C & -(D + D^\top) \end{pmatrix} \preceq 0$$

for some matrix $P = P^\top \succ 0$. It is called strictly passive provided that

$$\begin{pmatrix} A^\top P + PA + \varepsilon P & PB - C^\top \\ B^\top P - C & -(D + D^\top) \end{pmatrix} \preceq 0 \tag{2.2}$$

for some $\varepsilon > 0$ and matrix $P = P^\top \succ 0$.

The results in this section are extended from the passivity theorems that are formulated for the absolutely stability problems for Lur'e systems (see [3]).

Consider an input-free LCS given by

$$\begin{cases} \dot{x}(t) = Ax(t) + B\lambda(t), t \geq 0 \\ 0 \leq \lambda(t) \perp Cx(t) + D\lambda(t) \geq 0 \end{cases} \tag{2.3}$$

with (A, B, C, D) passive in the sense of Definition 1.1. Here A, B, C and D are real constant matrices with appropriate sizes, $x(t) \in \mathbb{R}^n$, $\lambda(t) \in \mathbb{R}^p$.

Lemma 2.1. *([9], Lemma 7.14) A state x^* is an equilibrium point of (2.3) if and only if there exists $\lambda^* \in \mathbb{R}^p$ satisfying*

$$\begin{cases} Ax^* + B\lambda^* = 0 \\ 0 \leq \lambda^* \perp Cx^* + D\lambda^* \geq 0. \end{cases} \tag{2.4}$$

Moreover, if A is invertible, then all equilibrium points of (2.3) are characterized by

$$x^* = -A^{-1}B\lambda^* \text{ and } 0 \leq (-CA^{-1}B + D)\lambda^* \perp \lambda^* \geq 0. \tag{2.5}$$

From Lemma 2.1 it follows that $x^* = 0$ is an equilibrium point. Next, we will deal with sufficient conditions for nature of equilibrium points.

Theorem 2.1. *([9], Theorem 7.15) Consider an LCS given by (2.3) such that (A, B, C, D) is passive, matrices $[B \quad AB \quad \dots \quad A^{n-1}B]$ and $[C^\top \quad A^\top C^\top \quad \dots \quad (A^\top)^{n-1}C]$ have full rank, and $\begin{pmatrix} B \\ D + D^\top \end{pmatrix}$ has full column rank. Then this LCS has only Lyapunov stable equilibrium points x^* given by (2.5). Moreover, if (A, B, C, D) is strictly passive, then $x^* = 0$ is the only equilibrium point, which is asymptotically stable.*

2.2 State feedback controller

As we know, Dissipativity theory gives a framework for the design and analysis of control systems using an input-output description based on energy-related considerations. The main idea behind this is that many important physical systems have certain input-output properties related to the conservation, dissipation and transport of energy [4]. Our problems in this topic are derived from linear electrical networks consisting of resistors, inductors, capacitors, and ideal diodes, thus it allows us for an approach to control systems design and analysis.

Now we focus on the linear complementarity system give by (2.1). First, we need to make some assumptions for our problem.

Assumption 2.1. There exists a multiplier λ_d such that desired trajectory x_d satisfies

$$\begin{cases} \dot{x}_d(t) = Ax_d(t) + B\lambda_d(t) + Eu_d(t) \\ 0 \le \lambda_d(t) \perp Cx_d(t) + D\lambda_d(t) + Fu_d(t) \ge 0, \end{cases}$$

for a given input $u_d \in L^1_{\text{loc}}(\mathbb{R}_+; \mathbb{R}^m)$.

Remark 2.1. For asymptotic tracking, $\|x(t) - x_d(t)\| \to 0$, as $t \to \infty$, Assumption 2.1 is *necessary*. Let us consider the following problem

$$\begin{cases} \dot{x}(t) = -x(t) + \lambda(t) \\ 0 \le \lambda(t) \perp x(t) + \lambda(t) + u(t) \ge 0. \end{cases} \tag{2.6}$$

From Proposition 1.1, the system (2.6) has a unique solution for any initial condition $x(0) = x_0 \in \mathbb{R}$ due to $d = 1 > 0$. Our aim is to find u such that the solution $x(t) \to -a$, as $t \to \infty$ with $a > 0$. It is easy to check that Assumption 2.1 is violated. We will prove that there is no controller u satisfying this problem.

One can rewrite (2.6) as

$$\dot{x}(t) = \begin{cases} -x(t) & \text{if } x(t) + u(t) \ge 0 \\ -2x(t) - u(t) & \text{if } x(t) + u(t) \le 0. \end{cases}$$

It yields

$$x(t) = \begin{cases} C_1 e^{-t} & \text{if } x(t) + u(t) \ge 0 \\ C_2 e^{-2t} - \int_{t_0}^{t} e^{-2(t-s)} u(s) \mathrm{d}s & \text{if } x(t) + u(t) \le 0, \end{cases}$$

for some C_1 and C_2. Since the trajectory $x(t) \to -a < 0$, $t \to \infty$, the controller u must satisfy $x(t) + u(t) \le 0$ for large t. That means the second mode has to be activated. It follows that $\lim\limits_{t\to\infty} u(t) \le -\lim\limits_{t\to\infty} x(t) = a$. Since $x(t)$ has to converge to $-a$ when $t \to \infty$,

$$\lim_{t\to\infty} \int_{t_0}^{t} e^{-2(t-s)} u(s) \mathrm{d}s = a.$$

Using L'Hospital rule, one gets

$$\lim_{t\to\infty} \frac{u(t)}{2} = a.$$

We got a contradiction. Therefore, there is no controller u for this problem.

Note that we can solve the regulation problem $(x(t) \to 0)$. In fact, if we choose $u(t) = \mathbf{k}x(t)$, then from (2.6) we obtain

$$\begin{cases} \dot{x}(t) = -x(t) + \lambda(t) \\ 0 \leq \lambda(t) \perp (1+\mathbf{k})x(t) + \lambda(t) \geq 0. \end{cases} \tag{2.7}$$

Now, we choose $\mathbf{k}$ such that $(-1, 1, 1+\mathbf{k}, 1)$ is strictly passive, for instance $\mathbf{k} = 0$. Using Theorem 2.1, we claim that 0 is the only equilibrium point of (2.7), which is asymptotically stable.

Assumption 2.2. There exists a matrix K such that the quadruple

$$(A + EK, B, C + FK, D)$$

is strictly passive.

Lemma 2.2. *Suppose that (A, B, C, D) is (strictly) passive. Then, the system*

$$\begin{cases} \dot{x}(t) = Ax(t) + B\lambda(t) + Eu(t), \\ 0 \leq \lambda(t) \perp Cx(t) + D\lambda(t) + Fu(t) \geq 0, \\ x(0) = x_0. \end{cases} \tag{2.8}$$

has a unique global solution for any input $u \in L^1_{\text{loc}}(\mathbb{R}_+; \mathbb{R}^m)$. Moreover, the solution is continuous on $[0, +\infty[$.

Proof. We will only prove the lemma in case (A, B, C, D) is strictly passive. In case (A, B, C, D) is passive, the proof of the lemma is similar but $\varepsilon = 0$ in (2.2).

Let P and $\varepsilon > 0$ be a solution to $P = P^\top \succ 0$ and

$$Q := \begin{pmatrix} A^\top P + PA + \varepsilon P & PB - C^\top \\ B^\top P - C & -(D + D^\top) \end{pmatrix} \preceq 0.$$

Since $Q \preceq 0$, D is necessarily non-negative definite.

Case 1: $D = 0$. From Lemma 2.3 below, we have $PB - C^\top = 0$. Following the same lines as in Section 1.1.1, the system (2.8) may be written as a differential inclusion. Then the well-posedness of the solution is claimed by Theorem 1.1.

Case 2: $D = \begin{pmatrix} D_1 & 0 \\ 0 & 0 \end{pmatrix}$, where $D_1 \succ 0$ is square of dimension $q < p$. We can see that the Moore-Penrose pseudo inverse of the matrix $D + D^\top$ is

$$(D + D^\top)^+ := \begin{pmatrix} (D_1 + D_1^\top)^{-1} & 0 \\ 0 & 0 \end{pmatrix}.$$

Again, from Lemma 2.3, one obtains

$$(PB - C^\top)[I_p - (D + D^\top)^+(D + D^\top)] = 0. \tag{2.9}$$

By writing $B = \begin{pmatrix} B_1 & B_2 \end{pmatrix}$, $C = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$ where B_1, B_2, C_1, C_2 have appropriate sizes, then (2.9) is equivalent to

$$\begin{pmatrix} PB_1 - C_1^\top & PB_2 - C_2^\top \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & I_{p-q} \end{pmatrix} = 0.$$

That implies $PB_2 - C_2^\top = 0$. Following the lines in Section 1.1.2, we can conclude that the system (2.8) has a unique solution.

Case 3: $D \succ 0$. Using Proposition 1.1, the system (2.8) has a unique solution.

The continuity of the solution in such cases is also obtained since the input $u \in L^1_{\text{loc}}(\mathbb{R}_+; \mathbb{R}^m)$.

□

We are at the position to state the main result in this section. The following proposition gives us a form of controller u to solve our problems.

Proposition 2.1. *Suppose that Assumptions 2.1 and 2.2 hold. Then the closed-loop system (2.1) with the state feedback controller*

$$u(t) = K[x(t) - x_d(t)] + u_d(t)$$

has a unique global solution $x(\cdot)$, and $\|x(t) - x_d(t)\| \to 0$ as $t \to +\infty$.

Proof. With the state feedback controller $u = K(x - x_d) + u_d$, the closed-loop system (2.1) can be rewritten as

$$\begin{cases} \dot{x}(t) = (A + EK)x(t) + B\lambda(t) + E(-Kx_d(t) + u_d(t)), \\ 0 \le \lambda(t) \perp (C + FK)x(t) + D\lambda(t) + F(-Kx_d(t) + u_d(t)) \ge 0, \\ x(0) = x_0. \end{cases} \tag{2.10}$$

The well-posedness can be claimed by using Lemma 2.2 because the quadruple $(A+EK, B, C+FK, D)$ is strictly passive. We just need to show that

$$\|x(t;x_0) - x_d(t)\| \to 0, \text{ as } t \to +\infty.$$

Set $z \triangleq x - x_d$. From Assumptions 2.1 and 2.2, we obtain

$$\dot{z}(t) = Az(t) + B[\lambda(t) - \lambda_d(t)] + E[u(t) - u_d(t)]. \tag{2.11}$$

Using the state feedback controller $u(t) = K[(x(t) - x_d(t)] + u_d(t)$, we obtain

$$\dot{z}(t) = (A+EK)z(t) + B(\lambda(t) - \lambda_d(t)).$$

From Assumption 2.2, let P and $\varepsilon > 0$ be a solution to $P = P^\top \succ 0$ and

$$Q := \begin{pmatrix} (A+EK)^\top P + P(A+EK) + \varepsilon P & PB - (C+FK)^\top \\ B^\top P - (C+FK) & -(D+D^\top) \end{pmatrix} \preceq 0.$$

Since $P = P^\top \succ 0$, it follows that $\langle Px, x\rangle \geq \lambda_1(P)\|x\|^2$ for all $x \in \mathbb{R}^n$, where $\lambda_1(P) > 0$ is the smallest eigenvalue of P.

Consider the Lyapunov function $V : \mathbb{R} \to \mathbb{R}_+$ defined as $V(z) = z^\top Pz$, then $V(\cdot)$ is continuously differentiable and its derivative along the trajectories of (2.11) satisfies the following for almost all $t \in [0, \infty)$:

$$\dot{V}(z(t)) = z(t)^\top[(A+EK)^\top P + P(A+EK)]z(t) + 2z(t)^\top PB[\lambda(t) - \lambda_d(t)].$$

To shorten notation, we denote $\bar{A} := A + EK, \bar{C} := C + FK$ and $z(t) := z(t; z_0)$. Then, we have

$$\begin{aligned} \dot{V}(z(t)) &= z(t)^\top(\bar{A}^\top P + P\bar{A})z(t) + 2z(t)^\top PB[\lambda(t) - \lambda_d(t)], \\ &= z(t)^\top(\bar{A}^\top P + P\bar{A})z(t) + 2z(t)^\top(PB - \bar{C}^\top)[\lambda(t) - \lambda_d(t)] \\ &\quad + 2[\lambda(t) - \lambda_d(t)]^\top \bar{C}z(t). \end{aligned}$$

holds for any $t \geq 0$. From the complementarity conditions in Assumption 2.2, we have

$$\begin{aligned} 0 \leq \lambda(t) \perp Cx(t) + D\lambda(t) + Fu(t) \geq 0, \\ 0 \leq \lambda_d(t) \perp Cx_d(t) + D\lambda_d(t) + Fu_d(t) \geq 0. \end{aligned}$$

Using the fact that $(u_1 - u_2)^\top(v_1 - v_2) \leq 0$ holds for $u_1, u_2, v_1, v_2 \in \mathbb{R}^p$ satisfying

$0 \le u_1 \perp v_1 \ge 0$ and $0 \le u_2 \perp v_2 \ge 0$, one obtains

$$[\lambda(t) - \lambda_d(t)]^\top [C(x(t) - x_d(t)) + D\,(\lambda(t) - \lambda_d(t)) + F(u(t) - u_d(t))] \le 0.$$

or equivalently,

$$[\lambda(t) - \lambda_d(t)]^\top [\bar{C} z(t) + D\,(\lambda(t) - \lambda_d(t))] \le 0.$$

It follows that

$$\begin{aligned}\dot{V}(z(t)) &\le z(t)^\top (\bar{A}^\top P + P\bar{A}) z(t) + 2z(t)^\top (PB - \bar{C}^\top)[\lambda(t) - \lambda_d(t)] \\ &\quad - [\lambda(t) - \lambda_d(t)]^\top (D + D^\top)[\lambda(t) - \lambda_d(t)] \\ &= (z(t) \quad \lambda(t) - \lambda_d(t))^\top Q \begin{pmatrix} z(t) \\ \lambda(t) - \lambda_d(t) \end{pmatrix} - \varepsilon z(t)^\top P z(t).\end{aligned}$$

Since $Q \preceq 0$,

$$\dot{V}(z(t)) \le -\varepsilon \langle Pz(t), z(t) \rangle \le -\varepsilon \lambda_1(P) \|z(t)\|^2.$$

Thus

$$\langle Pz(t), z(t) \rangle \le \langle Pz_0, z_0 \rangle - \varepsilon \lambda_1(P) \int_0^t \|z(s)\|^2 \mathrm{d}s.$$

Hence

$$\|z(t)\|^2 \le \frac{\langle Pz_0, z_0 \rangle}{\lambda_1(P)} - \varepsilon \int_0^t \|z(s)\|^2 \mathrm{d}s.$$

From Gronwall's lemma, one deduces that

$$\|z(t)\|^2 \le \frac{\langle Pz_0, z_0 \rangle}{\lambda_1(P)} e^{-\varepsilon t}.$$

That implies $\|z(t)\| \to 0$ or equivalently $\|x(t) - x_d(t)\| \to 0$ as $t \to \infty$.

□

In summary, controller u stated in Proposition 2.1 depends on tracking error $(x - x_d)$ and input u_d and feedback gain K, while K depends on the matrices A, B, C, D, E and F, the coefficient matrices of the problem. This is depicted in Figure 2.1.

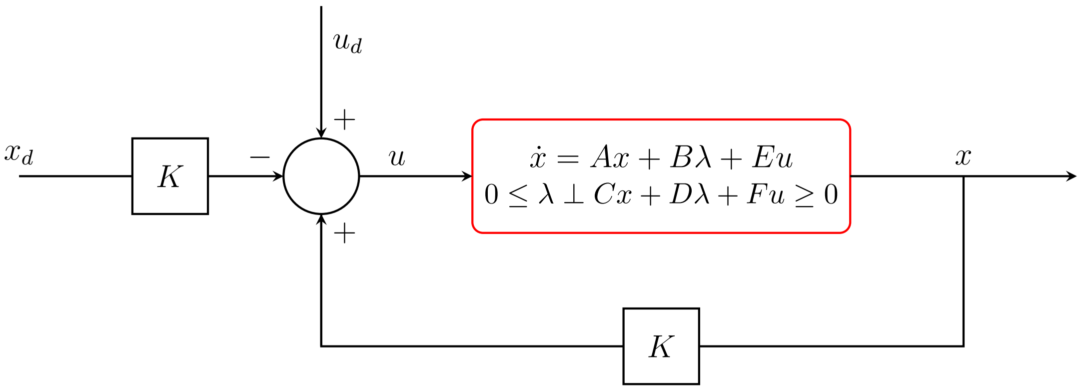


Figure 2.1: LCS with state feedback.

2.3 Determine gain K

2.3.1 Linear Matrix Inequality (LMI) Problems

To complete the design of the controller u, it is necessary to find K that ensures that the matrix inequality

$$\begin{pmatrix} (A+EK)^\top P + P(A+EK) + \varepsilon P & PB - (C+FK)^\top \\ B^\top P - (C+FK) & -(D+D^\top) \end{pmatrix} \preceq 0 \tag{2.12}$$

has a solution $P = P^\top \succ 0$ for some $\varepsilon > 0$.

Lemma 2.3 ([4], Lemma A.65)**.** *Let us assume that Q and R are symmetric. Then $\begin{pmatrix} Q & S \\ S^\top & R \end{pmatrix} \succeq 0$ if and only if $R \succeq 0, Q - SR^+S^\top \succeq 0$, and $S(I - R^+R) = 0$. Here R^+ is the Moore-Penrose pseudo inverse of the matrix R.*

For problems with small dimensions, Lemma 2.3 can be used to find a consistent matrix K by solving a system of equations and inequations. However, for higher dimension, we must have a useful tool to treat the problems. To do this, (2.12) needs to be reformulated to an LMI problem which is solved by a powerful software, MATLAB.

Let us multiply the left-hand side of (2.12) by $\begin{pmatrix} Q & 0 \\ 0 & I \end{pmatrix}$, where Q is the inverse matrix of P. Then, we have

$$\begin{pmatrix} Q(A+EK)^\top + (A+EK)Q + \varepsilon Q & B - Q(C+FK)^\top \\ B^\top - (C+FK)Q & -(D+D^\top) \end{pmatrix} \preceq 0. \tag{2.13}$$

Let us define a second new variable $L = KQ$. It follows that

$$\begin{pmatrix} QA^\top + AQ + L^\top E^\top + EL + \varepsilon Q & B - QC^\top - L^\top F^\top \\ B^\top - CQ - FL & -(D+D^\top) \end{pmatrix} \preceq 0. \tag{2.14}$$

This gives an LMI feasibility problem in the new variables $Q = Q^T \succ 0$ and L. After solving this LMI, the feedback gain K can be recovered from $K = LQ^{-1}$.

2.3.2 Solving LMI using MATLAB Toolboxes

LMI Toolbox of MATLAB is a powerful toolbox providing a set of useful functions to solve LMIs. To solve an LMI problem in MATLAB, there are 3 steps.

Step 1. Initializing the LMI description.

Step 2. Defining the Decision Variables.

Step 3. Defining the LMIs one by one.

In this section, we will show the code to get feedback gain K by solving the LMI given in (2.14). The code is given in Section B.1 in Appendix. For more details about the script, we refer the reader to [7] or visit https://fr.mathworks.com/help/robust/lmis.html.

Another toolbox that may be helpful to solve LMI problems is YALMIP. It is available freely to be used by researchers. Like LMI toolbox, to solve an LMI problem using YALMIP requires 3 steps:

Step 1. Defining Decision Variables.

Step 2. Defining Constraints.

Step 3. Solving Optimization Problems.

For more information please log on to the following: https://yalmip.github.io.

2.4 Numerical simulation

2.4.1 Numerical scheme

The numerical simulation for our problems is based on the Moreau-Jean's time-stepping scheme which is reported in [1].

We denote by $0 = t_0 < t_1 < \cdots < t_k < \cdots < t_N = T$ a finite partition of the time interval $[0, T]$ $(T > 0)$. The integer N stands for the number of time intervals in the partition. The length of a time step is denoted by $h_k = t_{k+1} - t_k$. For simplicity sake, we only consider a constant time length $h = h_k$ $(0 \leq k \leq N-1)$.

Let us consider the dynamic system (2.1) with state feedback controller $u = K(x - x_d) + u_d$. Then,

$$\begin{cases} \dot{x}(t) = (A + EK)x(t) + B\lambda(t) - EKx_d(t) + Eu_d(t), \\ 0 \leq \lambda(t) \perp (C + FK)x(t) + D\lambda(t) + F(-Kx_d(t) + u_d(t)) \geq 0. \end{cases} \tag{2.15}$$

Integrating the above ODE over a time step $[t_k, t_{k+1}]$ of length h, it follows

$$\int_{t_k}^{t_{k+1}} \dot{x}(t)\mathrm{d}t = \int_{t_k}^{t_{k+1}} (A+EK)x(t)\mathrm{d}t + \int_{t_k}^{t_{k+1}} B\lambda(t)\mathrm{d}t + \int_{t_k}^{t_{k+1}} [Eu_d(t) - EKx_d(t)]\mathrm{d}t.$$

The left-hand side is equal to $x(t_{k+1}) - x(t_k)$. The right-hand terms are approximated by using the $\theta-$method and an implicit Euler integration. Specifically,

$$\begin{aligned}
\int_{t_k}^{t_{k+1}} (A+EK)x(t)\mathrm{d}t &\approx h(A+EK)[\theta x(t_{k+1}) + (1-\theta)x(t_k)],\\
\int_{t_k}^{t_{k+1}} Eu_d(t)\mathrm{d}t &\approx hE[\theta u_d(t_{k+1}) + (1-\theta)u_d(t_k)],\\
\int_{t_k}^{t_{k+1}} -EKx_d(t)\mathrm{d}t &\approx -hEK[\theta x_d(t_{k+1}) + (1-\theta)x_d(t_k)],\\
\int_{t_k}^{t_{k+1}} B\lambda(t)\mathrm{d}t &\approx hB\lambda(t_{k+1}),
\end{aligned}$$

for some $\theta \in [0,1]$. By replacing the accurate solution $x(t_k)$ by the approximated value x_k, we obtain

$$\begin{aligned}
[I - h(A+EK)\theta]x_{k+1} =& [I + h(A+EK)(1-\theta)]x_k + hE[\theta u_d(t_{k+1}) + (1-\theta)u_d(t_k)]\\
&- hEK[\theta x_d(t_{k+1}) + (1-\theta)x_d(t_k)] + hB\lambda_{k+1}. \qquad (2.16)
\end{aligned}$$

Then, the time discretization of the non-smooth law is given by

$$0 \le \lambda_{k+1} \perp (C+FK)x_{k+1} + D\lambda_{k+1} + F(-Kx_d(t_{k+1}) + u_d(t_{k+1})) \ge 0.$$

Suppose that $[I - h(A+EK)\theta]$, denoted by W, is invertible. By substituting x_{k+1} from (2.16) to time discretization of the non-smooth law, one gets the LCP:

$$0 \le \lambda_{k+1} \perp M\lambda_{k+1} + q_{k+1} \ge 0,$$

in which

$$\begin{aligned}
M &= h(C+FK)W^{-1}B + D,\\
q_{k+1} &= (C+FK)W^{-1}[\bar{W}x_k + hE(u_{d,k+\theta} - Kx_{d,k+\theta})] + F(-Kx_{d,k+1} + u_{d,k+1}),\\
\bar{W} &= I + h(A+EK)(1-\theta),\\
u_{d,k+\theta} &= \theta u_d(t_{k+1}) + (1-\theta)u_d(t_k),\\
x_{d,k+\theta} &= \theta x_d(t_{k+1}) + (1-\theta)x_d(t_k).
\end{aligned}$$

Thus, h, θ can be chosen such that M is a positive definite matrix or a P-matrix in order to ensure that the time discretization of the non-smooth law possesses a solution λ_{k+1}.

2.4.2 SICONOS Platform

The SICONOS Platform is a scientific computing software dedicated to modeling, simulation, control, and analysis of nonsmooth dynamical systems. It is developed in the Tripop team-project at INRIA (The French National Institute for Research in Computer Science and Control) in Grenoble and distributed under GPL GNU license [1]. For more information, the reader can visit https://nonsmooth.gricad-pages.univ-grenoble-alpes.fr/siconos/index.html.

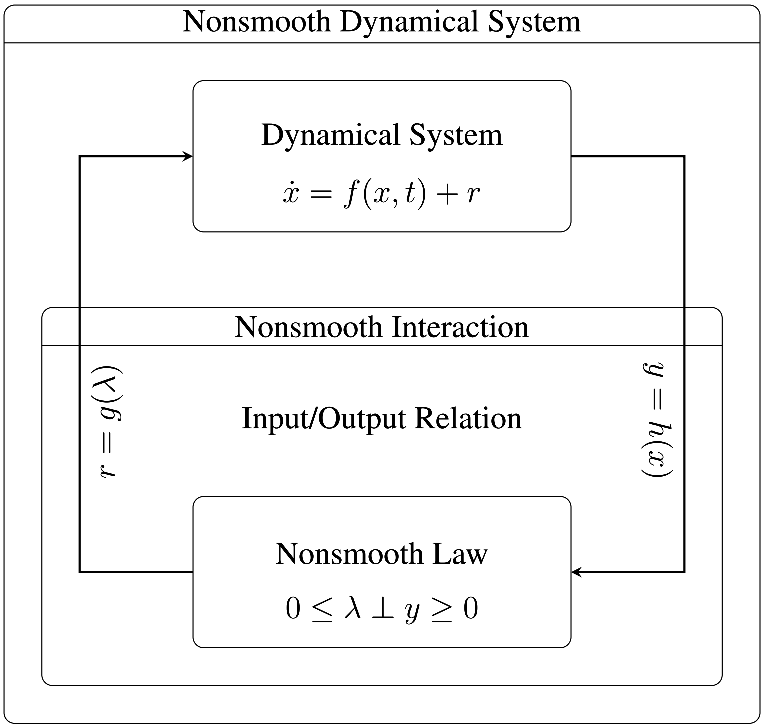


Figure 2.2: Nonsmooth Dynamical System Modeling Principle [1].

The SICONOS software is mostly written in C++ and it aims at providing a general and common tool for nonsmooth problems in various scientific fields such as applied mathematics, mechanics, robotics, electrical circuits, and so on. The general writing process for a problem treated with SICONOS includes two steps.

Step 1. Building a Nonsmooth Dynamical System

Step 2. Defining Simulation Strategy

All steps are built relying on its general principles of modeling and simulation. SICONOS Nonsmooth Dynamical System Modeling Principle is shown in Figure 2.2 (see also [1]).

2.4.3 Examples in scalar case

This part is dedicated to a short presentation of the general writing process for some examples, with $x(t) \in \mathbb{R}$, i.e., $n = 1, m = 1, p = 1$.

Let us consider a dynamical system given by

$$\begin{cases} \dot{x}(t) = -x(t) + \lambda(t) + u(t), \\ 0 \leq \lambda(t) \perp x(t) + \lambda(t) + u(t) \geq 0. \end{cases} \tag{2.17}$$

with initial state $x(0) = 1$. Our goal is to find a controller u such that $x(t) \to x_d(t)$ as $t \to +\infty$, where x_d satisfies

$$\begin{cases} \dot{x}_d(t) = -x_d(t) + \lambda_d(t) + u_d(t), \\ 0 \leq \lambda_d(t) \perp x_d(t) + \lambda_d(t) + u_d(t) \geq 0. \end{cases} \tag{2.18}$$

for given input $u_d(t) = \sin(t), t \geq 0$ and $x_d(0) = 0$.

According to Proposition 1.1, we claim that x_d is unique. We try to find a matrix (scalar) $\mathbf{k}$ such that $(-1 + \mathbf{k}, 1, 1 + \mathbf{k}, 1)$ is strictly passive. Then we have

$$\begin{pmatrix} 2(\mathbf{k} - 1)\mathbf{p} + \varepsilon\mathbf{p} & \mathbf{p} - \mathbf{k} - 1 \\ \mathbf{p} - \mathbf{k} - 1 & -2 \end{pmatrix} \preceq 0$$

for some $\mathbf{p} > 0$ and $\varepsilon > 0$. From Lemma 2.3, we get $2(\mathbf{k} - 1)\mathbf{p} + \varepsilon\mathbf{p} \leq 0$. This gives $2(\mathbf{k} - 1) + \varepsilon \leq 0$, or equivalently $\mathbf{k} \leq 1 - \varepsilon/2$. For instance, we can choose $u = 0.25(x - x_d) + u_d$ as a solution for our problem.

To get a simulation of this problem on SICONOS platform, we use the script as in Section B.2 in Appendix. In the following paragraphs, we turn our attention to Nonsmooth Dynamical System related and Simulation-related components in the platform which are used in our problem.

- `FirstOrderLinearDS` class, which describes the linear dynamical systems of first order in form (coefficients may be time invariant or not)

$$\begin{cases} \dot{x}(t) = A(t, z)x(t) + b(t, z) + r, \\ x(t_0) = x_0. \end{cases}$$

- A subclass of `FirstOrderLinearDS` class, `FirstOrderLinearTIDS`, which describes the linear dynamical systems of the form

$$\begin{cases} \dot{x}(t) = Ax(t) + b + r, \\ x(t_0) = x_0. \end{cases}$$

- `FirstOrderLinearTIR` class, which describes the first-order linear and time invariant relations:

$$\begin{cases} y = Cx + Fz + D\lambda + e, \\ r = B\lambda. \end{cases}$$

- `ComplementarityConditionNSL` class models a complementarity condition as

$$0 \leq y \perp \lambda \geq 0.$$

- `LCP` class represents the linear complementarity problem

$$\begin{cases} w = Mz + q, \\ 0 \leq w \perp z \geq 0. \end{cases}$$

Numerical simulation of Problem 2.17 with time-step $h = 5.10^{-4}$ and $\theta = 0.5$ is shown in Figure 2.3.

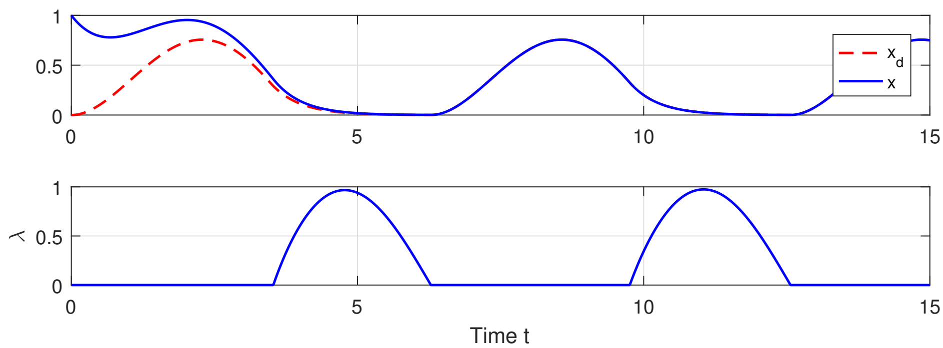


Figure 2.3: SICONOS simulation results for Problem 2.17

Now, we consider more dynamical systems by changing the constant coefficients in (2.17). That is,

$$\begin{cases} \dot{x}(t) = -x(t) + \lambda(t) + eu(t), \\ 0 \leq \lambda(t) \perp x(t) + d\lambda(t) + fu(t) \geq 0, \end{cases} \tag{2.19}$$

with d, e, f are real constants. By studying the same inputs $u_d(t) = \sin(t), t \geq 0$ and only changing d, e, f, we consider 5 different problems whose numerical results are shown in the following figures.

Notice that in the first two problems, the initial states can be chosen arbitrary due to $d > 0$, but three last problems (shown in Figures 2.6 - 2.8), the initial states ensure that $x(0) \geq 0$.

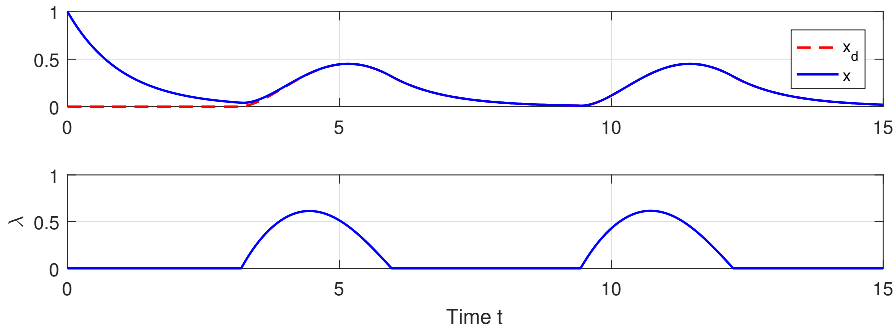


Figure 2.4: LCS (2.19) with $d = 1, e = 0, f = 1$

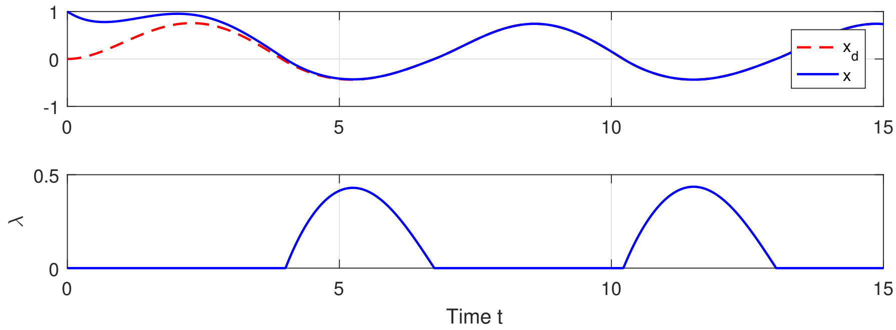


Figure 2.5: LCS (2.19) with $d = 1, e = 1, f = 0$

As we can see, in all problems, the state converges to desired trajectory perfectly. Moreover, there are two modes of the complementarity conditions activated in each problems, indicated by the value of the multiplier λ. Especially, in Figure 2.8, the complementarity condition for this problem is $0 \leq \lambda \perp x \geq 0$. From the plot in Figure 2.8, relation between x and λ is a complementarity one.

2.4.4 Example in higher dimension

Let us consider the circuit with two ideal diodes in Figure 2.9, with $R_1, R_2, R_3 > 0$, $L_1, L_2 > 0$, which is obtained from the circuit of Figure 2.3 in [1] by dropping the capacitor. The state variables are x_1, x_2, where x_1 is the current across the inductor L_1 and x_1 is the current across the inductor L_2. λ_1, λ_2 are the voltage of the diodes D_1 and D_2, respectively. The dynamics is given by

$$\begin{cases} \dot{x_1}(t) = -\frac{R_1+R_3}{L_1}x_1(t) + \frac{R_3}{L_1}x_2(t) + \frac{\lambda_1(t)}{L_1} + \frac{\lambda_2(t)}{L_1} \\ \dot{x_2}(t) = \frac{R_3}{L_2}x_1(t) - \frac{R_2+R_3}{L_2}x_2(t) - \frac{\lambda_2(t)}{L_2} + \frac{u(t)}{L_2}, \\ 0 \leq \begin{pmatrix} \lambda_1(t) \\ \lambda_2(t) \end{pmatrix} \perp \begin{pmatrix} x_1(t) \\ x_1(t) - x_2(t) \end{pmatrix} \geq 0. \end{cases} \tag{2.20}$$

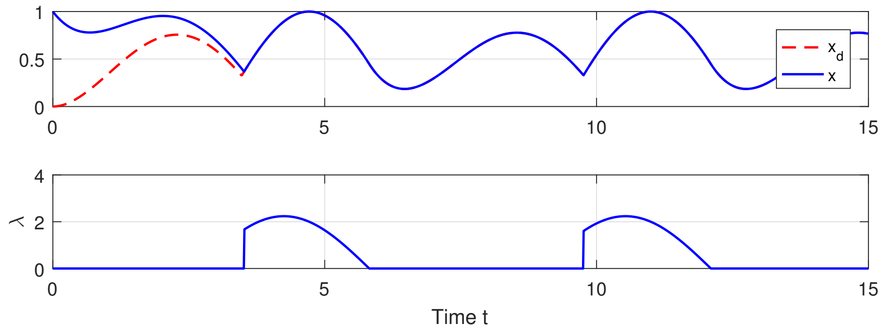


Figure 2.6: LCS (2.19) with $d = 0, e = 1, f = 1$

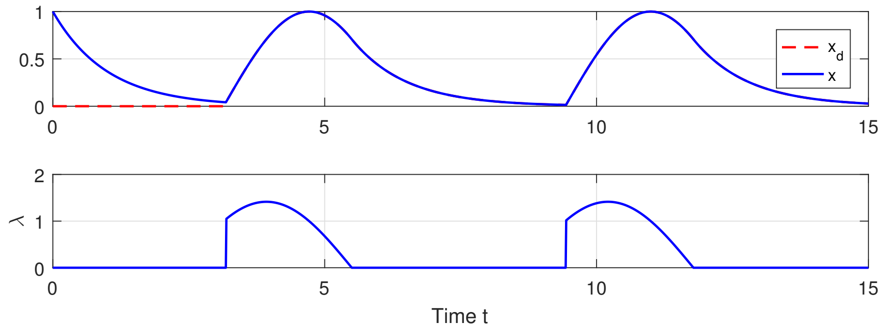


Figure 2.7: LCS (2.19) with $d = 0, e = 0, f = 1$

For instance, we choose $R_1 = R_2 = R_3 = 1\Omega$ and $L_1 = L_2 = 1\text{H}$. Then, the system can be written compactly as

$$\begin{cases} \dot{x}(t) = Ax(t) + B\lambda(t) + Eu(t), \\ 0 \leq \lambda(t) \perp Cx(t) \geq 0, \end{cases} \tag{2.21}$$

with

$$A = \begin{pmatrix} -2 & 1 \\ 1 & -2 \end{pmatrix}, B = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix}, \text{ and } E = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

We can see that the relation $PB = C^\top$ holds with $P = I$. From Section 1.1.1, (2.21) has a unique solution for any initial state $x(0) = x_0$ such that $Cx_0 \geq 0$.

Now, let us find controller u such that $\|x(t) - x_d(t)\| \to 0$, as $t \to +\infty$, where x_d is the desired trajectory, given as the solution of the LCS:

$$\begin{cases} \dot{x}_d(t) = Ax_d(t) + B\lambda_d(t) + Eu_d(t), \\ 0 \leq \lambda_d(t) \perp Cx_d(t) \geq 0, \end{cases}$$

for given input $u_d(t) = \sin(t), t \geq 0$. The numerical simulation of the desired trajectory is shown in Figure 2.10 with time step $h = 10^{-4}$ and $\theta = 0.5$. From Proposition 2.1,

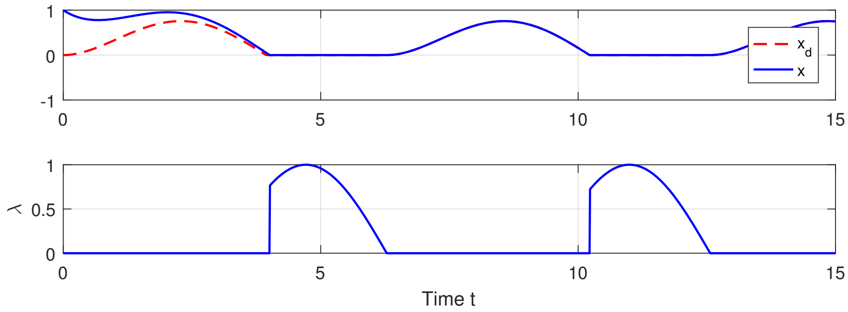


Figure 2.8: LCS (2.19) with $d = 0, e = 1, f = 0$

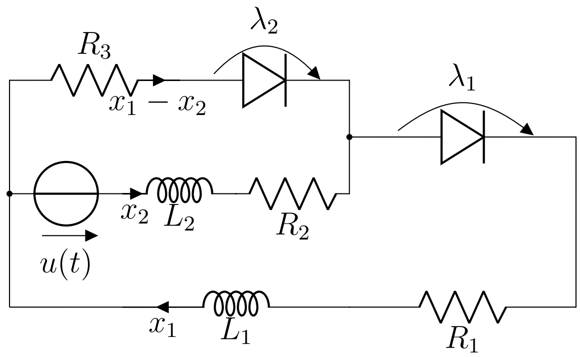


Figure 2.9: A circuit with two ideal diodes and a voltage source

we will find a controller u in the form $u = K(x - x_d) + u_d$ such that $(A + EK, B, C, 0)$ is strictly passive. Using YALMIP to solve this LMI problem. This gives

$$Q = \begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{pmatrix}, L = (-2.0, -1.0), \varepsilon = 4.$$

Then we can choose the matrix $K = (-2.0, -1.0)$ as a feedback gain for our problem. Proceeding with simulation of this problem in SICONOS, the results we obtained are depicted in Figure 2.11.

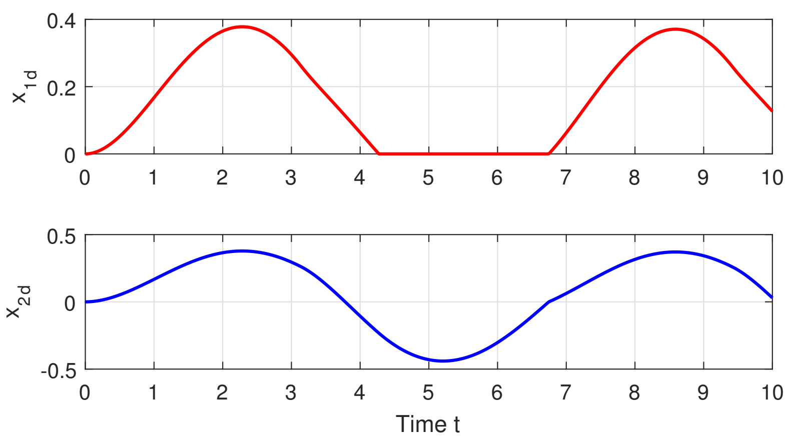


Figure 2.10: Desired trajectory simulated by SICONOS with $h = 10^{-4}, \theta = 0.5$

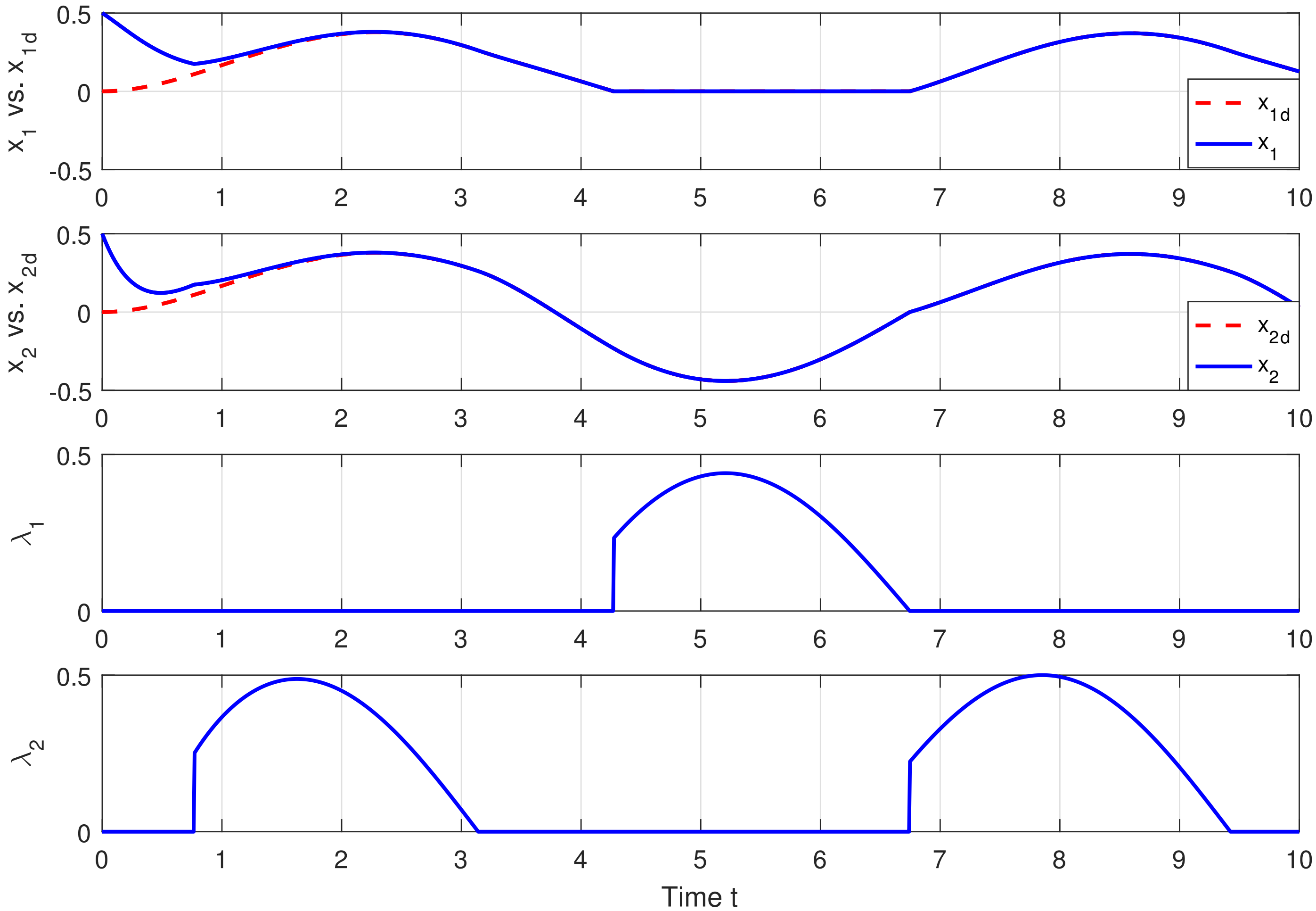


Figure 2.11: Simulation results by using SICONOS with $h = 10^{-4}$ and $\theta = 0.5$.

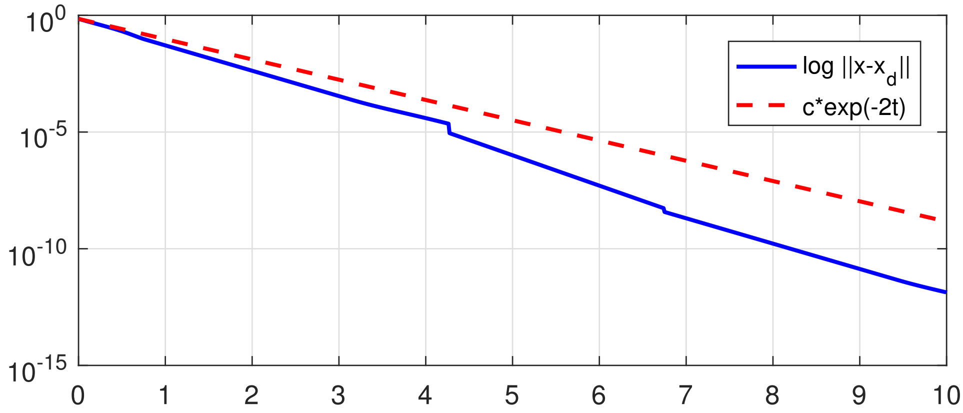


Figure 2.12: A semi-log plot of the error between x and x_d in time t ($c = \frac{\sqrt{2}}{2}$)

Notice that $(A + EK, B, C, 0)$ is strictly passive, it means

$$\begin{pmatrix} (A+EK)^\top P + P(A+EK) + \varepsilon P & PB - C^\top \\ B^\top P - C & 0 \end{pmatrix} \preceq 0 \tag{2.22}$$

holds for some $P = P^\top \succ 0, \varepsilon > 0$. As the solution obtained by MATLAB, we have $P = Q^{-1} = I, \varepsilon = 4$. From the proof of Proposition 2.1, for all $t \geq 0$, we get

$$\|x(t) - x_d(t)\|^2 \leq \frac{\langle P(x(0) - x_d(0)), x(0) - x_d(0)\rangle}{\lambda_1(P)} e^{-\varepsilon t}$$

or

$$\|x(t) - x_d(t)\| \leq \frac{\sqrt{2}}{2} e^{-2t}.$$

Hence the curve $y = \frac{\sqrt{2}}{2} e^{-2t}$ is considered as an upper-bound of the tracking error $\|x(t) - x_d(t)\|$. From Figure 2.12, it is obvious that the inequality $\|x(t) - x_d(t)\| \leq \frac{\sqrt{2}}{2} e^{-2t}$ holds for all $t \geq 0$ and $\|x(t) - x_d(t)\|$ tends to 0 as expected. Finally, the simulation of the controller u is shown in Figure 2.13.

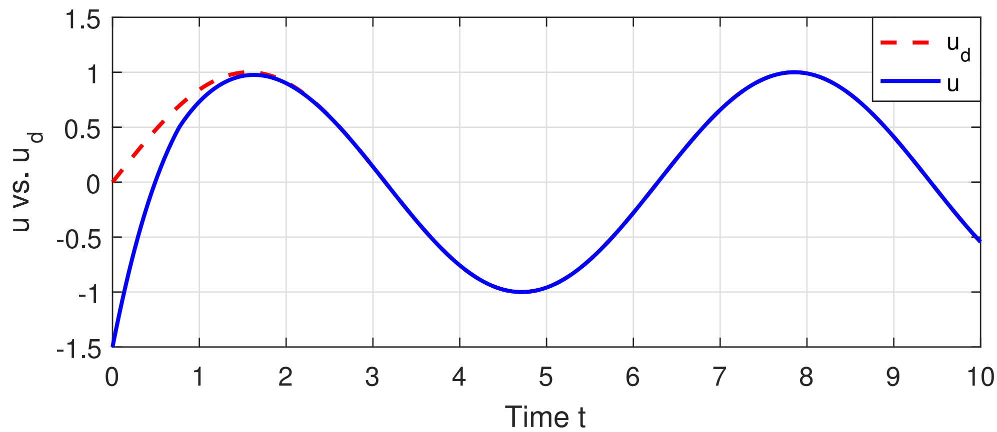


Figure 2.13: Simulation result of controller u

Remark 2.2. In Problem 2.21, 4 is the largest value of ε that ensures the LMI (2.22) is valid for some $P = P^\top \succ 0$ and K. In fact, we suppose that $K = (k_1, k_2)$. Since $D = 0$, it requires $PB = C^\top$. This relation holds if and only if $P = I$. Furthermore, we must ensure that

$$(A + EK)^\top P + P(A + EK) + \varepsilon P \preceq 0.$$

This gives us

$$\begin{pmatrix} -4+\varepsilon & k_1+2 \\ k_1+2 & 2(k_2-2)+\varepsilon \end{pmatrix} \preceq 0.$$

It follows that $-4 + \varepsilon \leq 0$, or equivalently $\varepsilon \leq 4$.

2.5 Extended controller form

2.5.1 Motivation

Let us consider the circuit that investigated in Example 1.1. The dynamics can be written compactly as

$$\begin{cases} \dot{x}(t) = Ax(t) + B\lambda(t) + Eu(t), \\ 0 \leq \lambda(t) \perp Cx(t) \geq 0 \end{cases} \tag{2.23}$$

with $A = \begin{pmatrix} 0 & 1 \\ -\frac{1}{\mathbf{LC}} & -\frac{\mathbf{R}}{\mathbf{L}} \end{pmatrix}, B = \begin{pmatrix} 0 \\ -\frac{1}{\mathbf{L}} \end{pmatrix}, C = \begin{pmatrix} 0 & -1 \end{pmatrix}, E = \begin{pmatrix} 0 \\ \frac{1}{\mathbf{L}} \end{pmatrix}$.
Suppose that the desired trajectory given as the solution of the LCS:

$$\begin{cases} \dot{x_d}(t) = Ax_d(t) + B\lambda_d(t) + Eu_d(t), \\ 0 \leq \lambda_d(t) \perp Cx_d(t) \geq 0, \end{cases}$$

for given input $u_d(t) = 1, t \geq 0$ and our aim is finding a controller u such that $\|x(t) - x_d(t)\| \to 0$, as $t \to +\infty$.

It is clearly that the relation $PB = C^\top$ holds with $P = \begin{pmatrix} p_1 & 0 \\ 0 & \mathbf{L} \end{pmatrix}$, for $p_1 > 0$. From results in Section 1.1.1, the desired trajectory x_d is unique. According to Proposition 2.1, if we try to find a controller u in the form $u = K(x - x_d) + u_d$, then the matrix K is chosen such that $(A + EK, B, C, 0)$ is strictly passive, i.e.,

$$\begin{pmatrix} (A+EK)^\top P + P(A+EK) + \varepsilon P & PB - C^\top \\ B^\top P - C & 0 \end{pmatrix} \preceq 0$$

holds for some $P = P^\top \succ 0$ and $\varepsilon > 0$. Since $D = 0$, we must have $PB = C^\top$. That implies $P = \begin{pmatrix} p_1 & 0 \\ 0 & \mathbf{L} \end{pmatrix}, p_1 > 0$. Suppose that $K = (k_1, k_2)$. Then, the matrix inequality

$$(A + EK)^\top P + P(A + EK) + \varepsilon P \preceq 0$$

is written as

$$\begin{pmatrix} \varepsilon p_1 & p_1 + k_1 - \frac{1}{\mathbf{C}} \\ p_1 + k_1 - \frac{1}{\mathbf{C}} & 2k_2 - 2\mathbf{R} \end{pmatrix} \preceq 0.$$

It follows that $\varepsilon p_1 \leq 0$. That is impossible because both ε and p_1 are positive real numbers.

One question has arisen. That is whether we can improve the form of the controller u to solve this problem. Fortunately, the answer is **yes**.

2.5.2 Form of the extended controller

Assumption 2.3. There exist matrices K, G such that the quadruple

$$(A + EK, B + EG, C + FK, D + FG)$$

is strictly passive and $D + FG$ is either a zero matrix, or a positive definite matrix, or a matrix in the form $\begin{pmatrix} D_1 & 0 \\ 0 & 0 \end{pmatrix}$ with $D_1 \succ 0$.

Proposition 2.2. *Suppose that Assumptions 2.1 and 2.3 hold. Then the closed-loop system (2.1) with the controller*

$$u(t) = K[x(t) - x_d(t)] + G[\lambda(t) - \lambda_d(t)] + u_d(t)$$

has a unique global solution $x(\cdot)$ and $\|x(t) - x_d(t)\| \to 0$ as $t \to +\infty$.

Proof. The proof is similar to the proof of Proposition 2.1. It suffices to replace B and D by $B + EG$ and $D + FG$ in our argument. □

As pointed out in Section 2.3, to complete the design of the controller u, we must determine matrices K, G such that

$$\begin{pmatrix} (A + EK)^\top P + P(A + EK) + \varepsilon P & P(B + EG) - (C + FK)^\top \\ (B + EG)^\top P - (C + FK) & -D - FG - (D + FG)^\top \end{pmatrix} \preceq 0 \quad (2.24)$$

has a solution $P = P^\top \succ 0$ for some $\varepsilon > 0$. Let us multiply the left-hand side of (2.24) by $\begin{pmatrix} Q & 0 \\ 0 & I \end{pmatrix}$, where $Q = P^{-1}$ and define a second new variable $L = KQ$. Then, we have

$$\begin{pmatrix} QA^\top + AQ + L^\top E^\top + EL + \varepsilon Q & B + EG - QC^\top - L^\top F^\top \\ B^\top + G^\top E^\top - CQ - FL & -D - FG - (D + FG)^\top \end{pmatrix} \preceq 0.$$

This is an LMI feasibility problem in the new variables $Q = Q^\top \succ 0$, G and L. The matrix K can be recovered from $K = LQ^{-1}$.

Example 2.1. Again, let us consider the circuit in Example 1.1. We have shown that there is no controller u in the form $u = K(x - x_d) + u_d$ that satisfies Proposition 2.1. For $\mathbf{R} = 10^2\Omega, \mathbf{L} = 10^{-6}\mathrm{H}, \mathbf{C} = 10^{-1}\mathrm{F}$, Assumption 2.3 holds with $\varepsilon = 0.0001$, $G = 0.999999$, and $K = (9.945811, -324.213572)$.

From Proposition 2.2, the controller $u = K(x - x_d) + G(\lambda - \lambda_d) + u_d$ solves our problem.

In summary, the closed-loop system structure with extended controller is depicted in Figure 2.14. It may be consider as an generalize of Figure 2.1.

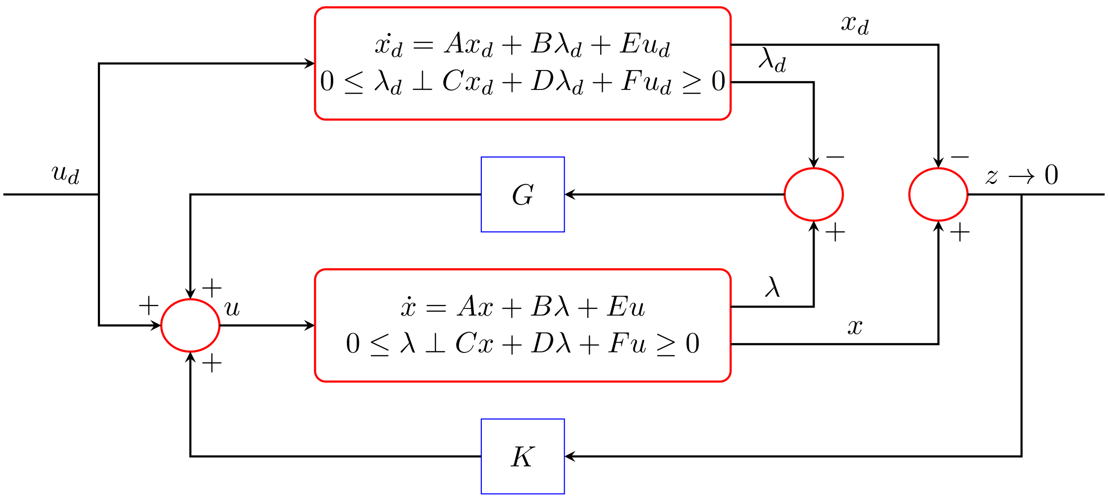


Figure 2.14: Closed-loop system structure with extended controller

2.5.3 Comments

Proposition 2.1 and 2.2 give us the form of the controller u to tackle a number of problems concerning trajectory tracking design for LCS with continuous solutions. However, there are some systems for which both propositions fail. Here is such a system.

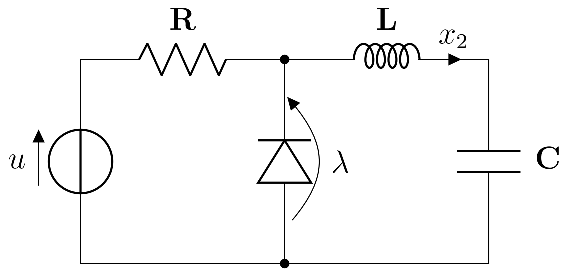


Figure 2.15: RLCD circuit with a voltage source.

Example 2.2. Let us investigate the circuit in Figure 2.15. The dynamics is given as

$$\begin{cases} \dot{x_1}(t) = x_2(t) \\ \dot{x_2}(t) = -\frac{1}{\mathbf{LC}}x_1(t) + \frac{\lambda(t)}{\mathbf{L}} \\ 0 \leq \lambda(t) \perp \frac{\lambda(t)}{\mathbf{R}} + x_2(t) - \frac{u(t)}{\mathbf{R}} \geq 0, \end{cases}$$

where x_1 is the charge on the capacitor **C**, x_2 is the current through the inductor **L**. The matrices A, B, C, D, E, F can be easily identified.

Suppose that there exists $K = (k_1, k_2)$ and G such that $(A + EK, B + EG, C + FK, D + FG)$, or equivalently $(A, B, C + FK, D + FG)$ is strictly passive. This means that

$$\begin{pmatrix} A^\top P + PA + \varepsilon P & PB - (C + FK)^\top \\ B^\top P - (C + FK) & -(D + FG) - (D + FG)^\top \end{pmatrix} \preceq 0$$

holds for some $P = P^\top \succ 0$ and $\varepsilon > 0$.

Assume that $P = \begin{pmatrix} p_1 & p_2 \\ p_2 & p_3 \end{pmatrix} \succ 0$. If G is chosen such that $D + FG = 0$, then $A^\top P + PA + \varepsilon P \preceq 0$. It follows that

$$\begin{pmatrix} -\frac{2p_2}{\mathbf{LC}} & p_1 - \frac{p_3}{\mathbf{LC}} \\ p_1 - \frac{p_3}{\mathbf{LC}} & 2p_2 \end{pmatrix} + \varepsilon \begin{pmatrix} p_1 & p_2 \\ p_2 & p_3 \end{pmatrix} \preceq 0.$$

This implies

$$-\frac{2p_2}{\mathbf{LC}} + \varepsilon p_1 \leq 0, \quad 2p_2 + \varepsilon p_3 \leq 0.$$

Thus, we get $\varepsilon p_1 + \frac{\varepsilon p_3}{\mathbf{LC}} \leq 0$, which is impossible since $p_1, p_3 > 0$.

If G satisfies $D + FG > 0$, then we must have

$$A^\top P + PA + \varepsilon P + [PB - (C + FK)^\top](D + FG)^+[B^\top P - (C + FK)] \preceq 0.$$

Note that $(D+FG)^+ = c > 0$. One deduces that

$$\begin{pmatrix} -\frac{2p_2}{\mathbf{LC}} & p_1 - \frac{p_3}{\mathbf{LC}} \\ p_1 - \frac{p_3}{\mathbf{LC}} & 2p_2 \end{pmatrix} + \varepsilon \begin{pmatrix} p_1 & p_2 \\ p_2 & p_3 \end{pmatrix} + c \begin{pmatrix} \frac{p_2}{\mathbf{L}} + \frac{k_1}{\mathbf{R}} \\ \frac{p_3}{\mathbf{L}} + \frac{k_2}{\mathbf{R}} - 1 \end{pmatrix} \begin{pmatrix} \frac{p_2}{\mathbf{L}} + \frac{k_1}{\mathbf{R}} & \frac{p_3}{\mathbf{L}} + \frac{k_2}{\mathbf{R}} - 1 \end{pmatrix} \preceq 0.$$

It follows that

$$-\tfrac{2p_2}{\mathbf{LC}} + \varepsilon p_1 + c(\tfrac{p_2}{\mathbf{L}} + \tfrac{k_1}{\mathbf{R}})^2 \leq 0, \tag{2.25}$$

$$2p_2 + \varepsilon p_3 + c(\tfrac{p_3}{\mathbf{L}} + \tfrac{k_2}{\mathbf{R}} - 1)^2 \leq 0. \tag{2.26}$$

From (2.25), (2.26), we obtain

$$0 < \varepsilon p_1 + \frac{\varepsilon}{\mathbf{LC}} p_3 + c(\frac{p_2}{\mathbf{L}} + \frac{k_1}{\mathbf{R}})^2 + \frac{c}{\mathbf{LC}}(\frac{p_3}{\mathbf{L}} + \frac{k_2}{\mathbf{R}} - 1)^2 \leq 0.$$

This contradicts our assumption since p_1, p_3, ε, c are positive numbers.

Therefore, there are not K and G such that $(A, B, C + FK, D + FG)$ is strictly passive. If we add a voltage source u_2 to the circuit in Figure 2.15 (see Figure 2.16), we

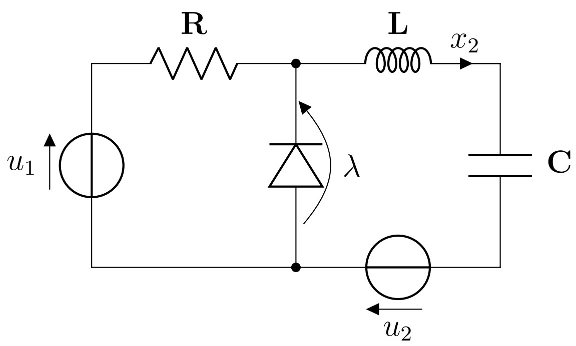


Figure 2.16: RLCD circuit with two voltage sources.

will have a new dynamics given by

$$\begin{cases} \dot{x_1}(t) = x_2(t) \\ \dot{x_2}(t) = -\frac{1}{\mathbf{LC}} x_1(t) + \frac{\lambda(t)}{\mathbf{L}} + \frac{u_2(t)}{\mathbf{L}} \\ 0 \leq \lambda(t) \perp \frac{\lambda(t)}{\mathbf{R}} + x_2(t) - \frac{u_1(t)}{\mathbf{R}} \geq 0. \end{cases}$$

Now, we have the LCS with $A = \begin{pmatrix} 0 & 1 \\ -\frac{1}{\mathbf{LC}} & 0 \end{pmatrix}$, $B = \begin{pmatrix} 0 \\ \frac{1}{\mathbf{L}} \end{pmatrix}$, $C = \begin{pmatrix} 0 & 1 \end{pmatrix}$, $D = \frac{1}{\mathbf{R}}$, $E = \begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{\mathbf{L}} \end{pmatrix}$, $F = \begin{pmatrix} -\frac{1}{\mathbf{R}} & 0 \end{pmatrix}$. For instance, let us consider $\mathbf{R} = \mathbf{C} = \mathbf{L} = 1$, given inputs $u_d = (u_{1,d}, u_{2,d})^\top = (1,1)^\top$ and initial state $x_0 = (0.5, 0.5)^\top$. From

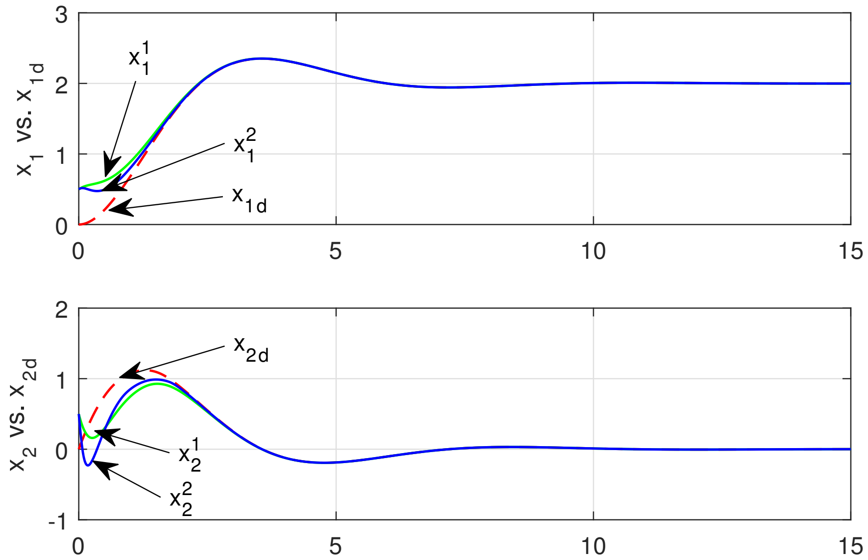


Figure 2.17: The desired trajectory and the state when using controller u^1 and u^2.

Proposition 2.1, we have the controller $u^1 = K(x - x_d) + u_d$ with

$$K = \begin{pmatrix} -1.554224 & -0.261066 \\ -3.228662 & -3.663074 \end{pmatrix}.$$

If we apply Proposition 2.2, we have the controller $u^2 = K(x - x_d) + G(\lambda - \lambda_d) + u_d$ with

$$K = \begin{pmatrix} -2.833123 & -1.041382 \\ -4.568759 & -3.291980 \end{pmatrix}, G = \begin{pmatrix} -0.500000 \\ 5.121218 \end{pmatrix}.$$

The matrices are obtained by using YALMIP with 6-digit accuracy. Using SICONOS to get the numerical results of this problem, the results are depicted in Figure 2.17 and 2.16. In both cases, we use explicit method ($\theta = 0$) with time step $h = 5.10^{-4}$.

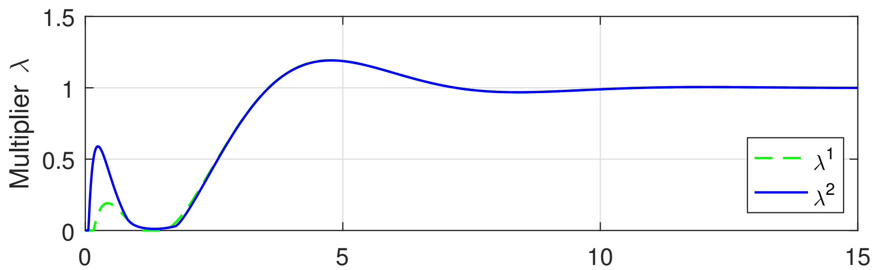


Figure 2.18: The multiplier λ^1 and λ^2 when using controller u^1 and u^2.

2.6 Decay rate of tracking error

As pointed out, for each trajectory tracking problem presented in the report, the feedback gain K is chosen satisfying Assumption 2.2 or Assumption 2.3 in the extended controller form. However, in general, such K is not unique. For instance, in Problem 2.17, Section 2.4.3, the matrix (scalar) $\mathbf{k}$ can be chosen such that $\mathbf{k} \leq 1-\varepsilon/2$ for $\varepsilon > 0$. It is clear that the set of feasible $\mathbf{k}$ is not unique even infinity. In some situations, we have more than one choice for K. Thus, a criterion has to be proposed with the aim at finding the "best" K in a set of feasible solutions.

From the proof of Proposition 1.1, for all $t \geq 0$, we have

$$\|x(t) - x_d(t)\|^2 \leq \frac{\langle P(x(0) - x_d(0)), x(0) - x_d(0)\rangle}{\lambda_1(P)} e^{-\varepsilon t}.$$

In which, P and $\varepsilon > 0$ are a solution to $P = P^\top \succ 0$ and

$$\mathcal{P}_\varepsilon := \begin{pmatrix} (A+EK)^\top P + P(A+EK) + \varepsilon P & PB - (C+FK)^\top \\ B^\top P - C - FK & -(D+D^\top) \end{pmatrix} \preceq 0.$$

The first idea in our mind is finding K such that ε as large as possible in order to increase decay rate of $\|x(t) - x_d(t)\|$. With the relation $\mathbf{k} \leq 1-\varepsilon/2$, for each $\varepsilon > 0$, there exists $\mathbf{k}$ that solves our problem. And $k \to -\infty$ provided that $\varepsilon \to +\infty$. Then, if we apply the controller $u = \mathbf{k}(x - x_d) + u_d$, it may be unbounded. That may be unsuitable in the case the controller has been bounded, for example, a voltage source with a limited voltage. Hence, we continue to find a consistent K ensures that it satisfies assumptions and LMI holds with the largest ε but $\|K\|$ is bounded by some given constant. Here, the norm of K is defined as the square root of the maximum eigenvalue of $K^\top K$.

Suppose that $\|K\| \leq \kappa$, for a given $\kappa > 0$. Our goal can be formulated under the form of an optimal problem $(\mathbf{P}_\kappa)$.

$$\begin{aligned} \max_{(\varepsilon,Q,K)} \quad & \varepsilon \\ \text{subject to} \quad & P = P^\top \succ 0, \varepsilon > 0, \\ & \mathcal{P}_\varepsilon \preceq 0, \|K\| \leq \kappa. \end{aligned}$$

Let us change variable $Q = P^{-1}$ and define a new varibale $L = KQ$. The constraint $\|K\| \leq \kappa$ can be replaced by sufficient conditions

$$Q \succeq I, \begin{pmatrix} \kappa I & L \\ L^\top & \kappa I \end{pmatrix} \succeq 0,$$

where I stands for identity matrices with appropriate sizes.

In fact, from $\begin{pmatrix} \kappa I & L \\ L^\top & \kappa I \end{pmatrix} \succeq 0$, it follows that $\kappa^2 I \succeq LL^\top$. That implies $\|L\| \leq \kappa$. Using $Q \succeq I$, we obtain

$$\|K\| = \|LQ^{-1}\| \leq \|L\|\|Q^{-1}\| \leq \kappa.$$

Now, the problem $(\mathbf{P}_\kappa)$ is completely rewritten as $(\mathbf{Q}_\kappa)$:

$$\begin{aligned} \max_{(\varepsilon,Q,L)} \quad & \varepsilon \\ \text{subject to} \quad & Q = Q^\top \succeq I, \varepsilon > 0, \\ & \mathcal{Q}_\varepsilon \preceq 0, \mathcal{L}_\kappa \succeq 0, \end{aligned}$$

in which

$$\mathcal{Q}_\varepsilon := \begin{pmatrix} QA^\top + AQ + L^\top E^\top + EL + \varepsilon Q & B - QC^\top - L^\top F^\top \\ B^\top - CQ - FL & -(D + D^\top) \end{pmatrix},$$

and

$$\mathcal{L}_\kappa := \begin{pmatrix} \kappa I & L \\ L^\top & \kappa I \end{pmatrix}.$$

Due to the product between ε and Q, the problem $(\mathbf{Q}_\kappa)$ cannot be solved directly using linear semidefinite programming (YALMIP).

Let us define a feasible set for each $\varepsilon > 0$ given by

$$\mathcal{F}_\varepsilon := \{(Q, L) : Q = Q^\top \succeq I, \mathcal{Q}_\varepsilon \preceq 0, \mathcal{L}_\kappa \preceq 0\}.$$

Then, Problem $(\mathbf{Q}_\kappa)$ is considered as finding the largest ε such that $\mathcal{F}_\varepsilon$ is non-empty.

Proposition 2.3. *Let $\varepsilon' > \varepsilon > 0$. Suppose that $\mathcal{F}_\varepsilon = \emptyset$. Then $\mathcal{F}_{\varepsilon'} = \emptyset$.*

Proof. Suppose that $\mathcal{F}_{\varepsilon'}$ is non-empty and $(Q, L) \in \mathcal{F}_{\varepsilon'}$. Then, we have

$$\mathcal{Q}_\varepsilon = \mathcal{Q}_{\varepsilon'} + (\varepsilon - \varepsilon') \begin{pmatrix} Q & 0 \\ 0 & 0 \end{pmatrix} \preceq 0.$$

That inequality holds because $\mathcal{Q}_{\varepsilon'} \preceq 0$ and $Q \succ 0$. This implies $(Q, L) \in \mathcal{F}_\varepsilon$, which contradicts our assumption. □

This property gives us a method to estimate the solution of the problem $(\mathbf{Q}_\kappa)$. The method called *Bisection Algorithm* can be described in a simple way as below. For more details, we refer the reader to https://yalmip.github.io/example/decayrate/.

1. Find a lower bound on optimal ε (any feasible ε you can compute).

2. Find an upper bound on optimal ε (increase the lower bound until a feasibility problem for fixed ε is infeasible).

3. Check value between lower and upper bound. If feasible, update lower bound, if infeasible update upper bound. Repeat until bounds are sufficiently close.

Now, by using YALMIP we definitely compute the optimal value of the problem $(\mathbf{Q}_\kappa)$. The code is given in Section B.3.

Example 2.3. Let us review the dynamical system which is surveyed in Section 2.4.4. The dynamics is written as

$$\begin{cases} \dot{x}(t) = Ax(t) + B\lambda(t) + Eu(t), \\ 0 \le \lambda(t) \perp Cx(t) \ge 0, \end{cases}$$

with

$$A = \begin{pmatrix} -2 & 1 \\ 1 & -2 \end{pmatrix}, B = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix}, \text{ and } E = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

If the feedback gain K do not require being bounded, then $K = (-2, -1)$ is a solution for our problem. But it is not a solution when we add a constraint in K, for instance $\|K\| \le \kappa$ with $\kappa = 2$. Among consistent matrices K, we can choose $K = (-2, 0)$ which makes ε to obtain the largest value ($\varepsilon = 4$) by solving Problem $(\mathbf{Q_2})$.

It is easy to see that the feasible set of $(\mathbf{Q}_\kappa)$ is larger and larger when we increase the κ and vice versa. Thus, the optimal value of $(\mathbf{Q}_\kappa)$ is non-decreased in κ. But it may be bounded (see Figure 2.19) or not, see for instance Problem 2.17.

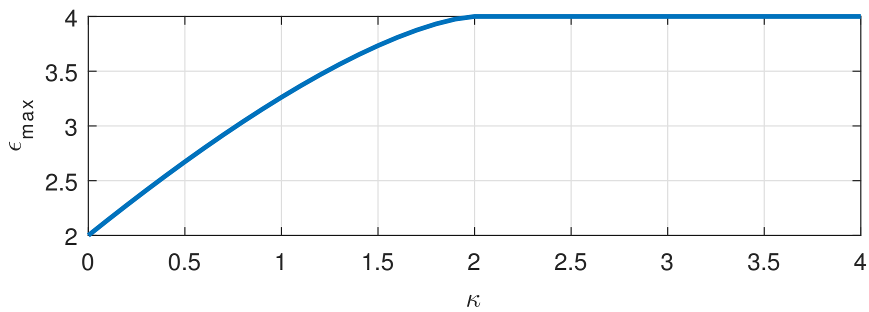


Figure 2.19: Relation between κ and optimal value of $(\mathbf{Q}_\kappa)$

For the problem $(\mathbf{Q}_4)$, we have at least two solutions. They are $K = k_1 = (-2, 0)$ and $K = k_2 = (-2, -1)$ with $\|k_1\| < \|k_2\| \le 4$. The question is among feasible K of $(\mathcal{Q}_4)$, which one does have a minimal norm? According to Figure 2.19, we can see that the minimal norm is approximate to 2, for example $K = (-2, 0)$. This leads us to study an interesting problem $(\mathbf{K})$:

$$\begin{aligned} \min_{(\kappa,Q,L)} \quad & \kappa \\ \text{subject to} \quad & Q = Q^\top \succeq I, \kappa \geq 0, \\ & \mathcal{Q}_{\varepsilon_{\max}} \preceq 0, \mathcal{L}_\kappa \succeq 0. \end{aligned}$$

Here, we denote by $\varepsilon_{\max}$ the optimal value of $(\mathbf{Q}_\kappa)$. Using YALMIP, we obtain the most suitable matrix K that solves our problem. The code can be found in Section B.3. Running that code, we get the best gain K for problem in Example 2.3 is $(-2, 0)$. Then, $\varepsilon_{\max} = 4$ and $\|K\| = 2$.

Remark 2.3. In the case we would like to use extended controller form, the results above may be used to analyse feed back gain K by replacing $\mathcal{Q}_\varepsilon$ by

$$\bar{\mathcal{Q}}_\varepsilon := \begin{pmatrix} QA^\top + AQ + L^\top E^\top + EL + \varepsilon Q & B + EG - QC^\top - L^\top F^\top \\ B^\top + G^\top E^\top - CQ - FL & -D - FG - (D + FG)^\top \end{pmatrix},$$

and we consider a new problem $(\bar{\mathbf{Q}}_\kappa)$:

$$\begin{aligned} \max_{(\varepsilon,Q,L)} \quad & \varepsilon \\ \text{subject to} \quad & Q = Q^\top \succeq I, \varepsilon > 0, \\ & \bar{\mathcal{Q}}_\varepsilon \preceq 0, \mathcal{L}_\kappa \succeq 0. \end{aligned}$$

Then, all previous steps can be redone and we have the most suitable K for our problem.

Conclusion

Besides reviewing several concepts of Convex analysis and the nature of equilibrium points of a dynamical system, during my internship, I have achieved several goals.

First of all, I have surveyed the existence and uniqueness of LCS in cases $D = 0$, $D \succeq 0$ and $D \succ 0$ and several examples in each case (see Examples 1.1, 1.2, 1.3). When $D = 0$ and $D \succeq 0$, with some suitable conditions (see Assumption 1.1, Theorem 1.1), we claim its well-posedness. For $D \succ 0$, according to Proposition 1.1, the LCS has a unique solution for any initial state x_0 and input u.

Secondly, I have investigated the multivalued Lur'e system. Some results about its well-posedness relying on Kato's Theorem and maximal monotonicity are stated as basis facts, they deal with a class of systems in which D is a nonzero feedthrough matrix, possibly positive semidefinite and non-symmetric. Dissipativity and stability results are background notions to construct a controller for a closed-loop, Proposition 1.6, Theorem 1.4.

For our main goal, we make some suitable assumptions and propose a form of state feedback controller $u = K(x - x_d) + u_d$ (see Proposition 2.1) based on Dissipativity theory to solve our problem. I suggest that we can use Dissipativity theory to establish a form of controller when the desired trajectory have state jumps.

Next, we consider an example that assumptions in Proposition 2.1 are not satisfied. From this, we give an extended form of the controller u to handle that example, see Proposition 2.2. We also deal with the problems that the controllers u are bounded. From that we propose a way to choose gain K that makes decay rate of tracking error to reach maximal value.

Finally, I used SICONOS Platform to model and simulate nonsmooth dynamical systems via examples in the scalar cases and higher dimension case. SICONOS is a useful tool to treat such problems. For the future works, we are going to work on:

- Shockley law versus complementarity conditions,
- robustness of controller with Shockley law with respect to Shockley law,
- robustness property with respect to uncertainties as A, B, C, D.
- advantage of an LCS approach versus an exponential one for control synthesis.

Appendix A

Introductory Material

This introductory chapter aims at providing some basic facts and notations concerning *convex analysis*, *convex functions*, and *equilibrium point* of a dynamic system, which will often be useful for us throughout this report. The material which follows is taken from [2].

A.1 Subdifferential and normal cones

A.1.1 Convex sets

Definition A.1 (Convex set)**.** A subset C of $\mathbb{R}^n$ is said to be *convex* if for each $x, y \in C$ and for each $\lambda \in [0, 1]$ we have $\lambda x + (1 - \lambda)y \in C$, i.e., the closed line segment $[x, y] \subset C$ whenever $x, y \in C$.

Definition A.2 (Cone)**.** A nonempty subset C of $\mathbb{R}^n$ is called a *cone* if for each $x \in C$ and each $\lambda \geq 0$ we have $\lambda x \in C$.

Definition A.3. Let C be a nonempty subset of $\mathbb{R}^n$.

(i) The *dual cone* C^* of C in $\mathbb{R}^n$ is defined by $C^* = \{p \in \mathbb{R}^n : \langle x, p\rangle \geq 0, \forall x \in C\}$.

(ii) The *polar cone* C° of C in $\mathbb{R}^n$ is defined by $C^\circ = \{p \in \mathbb{R}^n : \langle x, p\rangle \leq 0, \forall x \in C\}$.

Remark A.1. It is clear that $C^\circ = -C^*$ and we note that C^* is always a closed and convex cone.

Definition A.4 (Normal cone)**.** The *normal cone* to a nonempty closed convex subset C in $\mathbb{R}^n$ at a point $x \in C$ is defined by

$$\mathrm{N}_C(x) = \{p \in \mathbb{R}^n : \langle p, y - x\rangle \leq 0, \forall y \in C\}.$$

Definition A.5 (Tangent cone)**.** Let $C \in \mathbb{R}^n$ be a nonempty subset. A vector $d \in H$ is a *direction tangent* to C at a point $x \in C$ if there exists a sequence $(x_n)_{n\in\mathbb{N}} \subset C$ and a real sequence $(\tau_n)_{n\in\mathbb{N}}$ such that

$$x_n \to x, \tau \downarrow 0 \text{ and } \frac{x_n - x}{\tau_n} \to d \text{ as } n \to \infty.$$

The set of all such directions $d \in \mathbb{R}^n$ is called the *tangent cone* to C at $x \in C$, denoted by $\mathrm{T}_C(x)$.

A.1.2 Convex functions

An extended real-valued function is a function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$.

Definition A.6. Given a subset C of $\mathbb{R}^n$, we denote by $\psi_C(\cdot)$ the indicator function of C given by

$$\psi_C(x) = \begin{cases} 0 & \text{if } x \in C, \\ +\infty & \text{if } x \notin C. \end{cases}$$

Definition A.7. Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be an extended real-valued function.

(i) The effective domain of f is defined by $\mathrm{dom}(f) = \{x \in \mathbb{R}^n : f(x) < +\infty\}$. The function f is said to be proper if its effective domain is nonempty.

(ii) The epigraph of f is defined by $\mathrm{epi}(f) = \{(x, \lambda) \in \mathbb{R} \times \mathbb{R} : f(x) < \lambda\}$.

(iii) Given $\sigma \in \mathbb{R}$, the lower σ-level set of f is defined by

$$\mathrm{lev}_\sigma(f) = \{x \in \mathbb{R}^n : f(x) \leq \sigma\}.$$

Definition A.8. A proper function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is said to be convex if

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y), \tag{A.1}$$

for every $\lambda \in [0, 1]$ and every $x, y \in \mathrm{dom}(f)$.
If (A.1) holds with strict inequality for $\lambda \in (0, 1)$ and $x \neq y$, then f is called strictly convex.

Proposition A.1. *Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper function. Then*

$$f \textit{ is convex } \Leftrightarrow \mathrm{epi}(f) \textit{ is a convex set in } \mathbb{R} \times \mathbb{R}^n.$$

Proposition A.2. *Let $f_1, f_2 : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be two extended real-valued convex functions, let $\alpha_1 \geq 0, \alpha_2 \geq 0$ and $A : \mathbb{R}^n \to \mathbb{R}^n$ be an affine operator. The following functions are also convex:*

(i) $\alpha_1 f_1 + \alpha_2 f_2$,

(ii) $\sup\{f_1, f_2\}$,

(iii) $f_1 \circ A$ *and* $f_2 \circ A$.

Definition A.9. Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper function and let $x_0 \in \mathbb{R}^n$. We say that f is lower semicontinuous at x_0 if, for every sequence $(x_n)_{n\in\mathbb{N}}$ in $\mathbb{R}^n$ such that $x_n \to x_0$ as $n \to +\infty$, we have $\liminf\limits_{n\to+\infty} f(x_n) \geq f(x_0)$.

Proposition A.3. *Let* $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ *be a proper function. Then the following assertions are equivalent:*

(i) f *is lower semicontinuous,*

(ii) $\mathrm{epi}(f)$ *is closed in* $\mathbb{R}^n \times \mathbb{R}$,

(iii) $\mathrm{lev}_\sigma(f)$ *is closed in* $\mathbb{R}^n$ *for all* $\sigma \in \mathbb{R}$.

Example A.1. Let $C \subset \mathbb{R}^n$. The indicator function ψ_C is lower semicontinuous if and only if C is closed in $\mathbb{R}^n$.

Definition A.10. Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper and convex function. The subdifferential $\partial f(x)$ of f at $x \in \mathbb{R}^n$ is defined by

$$\partial f(x) = \{\langle p, y - x\rangle \leq f(y) - f(x) \text{ for all } y \in \mathbb{R}^n\}.$$

We associate to f a set-valued operator $\partial f : \mathbb{R}^n \rightrightarrows \mathbb{R}^n, x \mapsto \partial f(x)$. The domain of the operator ∂f, denoted by $\mathrm{Dom}(\partial f)$, is defined by

$$\mathrm{Dom}(\partial f) = \{x \in \mathbb{R}^n : \partial f(x) \neq \emptyset.\}$$

Definition A.11. Let $f : \mathbb{R}^n \to [-\infty, +\infty]$ be a function. The *conjugate* function $f^* : \mathbb{R}^n \to [-\infty, +\infty]$ associated with f is defined by

$$f^*(p) = \sup_{x\in\mathbb{R}^n} \{\langle p, x\rangle - f(x)\}.$$

This is known in the literature as the *Legendre-Fenchel transformation* or the *Fenchel conjugate*.

Proposition A.4. *Let* $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ *be a proper convex function. Then* $p \in \partial f(x)$ *if and only if* $f(x) + f^*(p) = \langle p, x\rangle$. *Moreover, if* f *is lower semicontinuous, then each of the above properties is equivalent to* $x \in \partial f^*(p)$, *which means that for every proper, lower semicontinuous and convex function* f *on* $\mathbb{R}^n$ *we get*

$$p \in \partial f(x) \Leftrightarrow x \in \partial f^*(p), \; i.e., \; (\partial f)^{-1} = \partial f^*.$$

A.2 Equilibrium point of a dynamic system

For an interval $I \subset \mathbb{R}$, and a function $f : I \to \mathbb{R}^n$, the variation of $f(\cdot)$ on the interval I is the supremum of $\sum_{i=1}^{k} \|f(x_i) - f(x_{i-1})\|$ over the set of of all finite sets of points $x_0 < x_1 < \cdots < x_k$ of I. When this supremum is finite, the function $f(\cdot)$ is said to be of *bounded variation* on I. An absolutely continuous (AC) function $f : I \to \mathbb{R}^n$ is a function that can be written as $f(t) = f(t_0) + \int_{t_0}^{t} \dot{f}(s)\mathrm{d}s$ for any $t_0, t \in I, t_0 \leq t$, and some $\dot{f} \in L^1(I, \mathbb{R}^n)$.

Let us consider a nonlinear system represented as

$$\begin{cases} \dot{x}(t) = f(x(t), t) \\ x(t_0) = x_0, \end{cases} \tag{A.2}$$

where $f(\cdot)$ is a nonlinear vector function, and $x \in \mathbb{R}^n$ is the state vector. We suppose that the system is well-posed, i.e., a unique solution exists globally.

Definition A.12 (Equilibrium)**.** A state $x^\star$ is an *equilibrium point* of (A.2) if $f(x^\star, t) = 0$ for all $t \geq t_0$.

Definition A.13 (Stable)**.** The equilibrium point $x = 0$ is stable at $t = t_0$ if for any $\rho > 0$ there exists an $r(\rho, t_0) > 0$ such that $\|x(t)\| < \rho$ for all $t \geq t_0$ whenever $\|x(t_0)\| < r$. Otherwise the equilibrium point $x = 0$ is unstable.

Definition A.14. The equilibrium point $x = 0$ is said to be:

- asymptotically stable at $t = t_0$ if it is stable and if it exists $r(t_0) > 0$ such that $\|x(t_0)\| < r(t_0) \Rightarrow x(t) \to 0$ as $t \to \infty$.
- exponentially stable if there exist two positive numbers α and λ such that $\|x(t)\| \leq \alpha \|x(t_0)\| e^{-\lambda(t-t_0)}, \ \forall t \geq t_0$, for $x(t_0)$ sufficiently small.
- globally asymptotically stable at $t = t_0$ if it is stable and $x(t) \to 0$ as $t \to \infty$ for all $x(t_0) \in \mathbb{R}^n$.

Appendix B

Scripts

B.1 MATLAB Code solving LMI problems

Assuming that the matrices A, B, C, D, E, and F already exist in the environment.

LMI Toolbox

```
% Step 1.
setlmis([])
% Step 2.
Q = lmivar(1,[size(A,1) 1]);
L = lmivar(2,[size(E,2) size(A,1)]);
epsilon = 0.001;
% Step 3.
% First LMI: inequation (3.9)
lmiterm([1 1 1 Q],A,1,'s');
lmiterm([1 1 1 L],E,1,'s');
lmiterm([1 1 1 Q],epsilon,1);
lmiterm([1 1 2 0],B);
lmiterm([1 1 2 Q],1,-C');
lmiterm([1 1 2 -L],1,-F');
lmiterm([1 2 2 0],-D-D');
% Second LMI: Q > 0
lmiterm([-2 1 1 Q],1,1);
LMISYS = getlmis;
% SOLVE
[~,xfeasp] = feasp(LMISYS);
Q = dec2mat(LMISYS,xfeasp,Q);
```

```
L = dec2mat(LMISYS,xfeasp,L);
%Feedback gain
K = L*Q^(-1);
```

YALMIP Toolbox

```
Q = sdpvar( size (A,1));
L = sdpvar( size (E,2), size (A,1));
% Set eps >0, eps2 >= 0
eps = 1;
eps2 = 1;
M = [Q*A'+A*Q+eps*P+L'*E'+E*L, B-CQ'-L'*F';
                 B'-C*Q-F*L,        -D-D'];
N = Q-eps2*eye(size(A,1));
 constraint =[M<=0,-N<=0];
 solvesdp( constraint );
Q = double(Q);
L = double(L);
% Feedback gain
K = L*Q^(-1);
```

B.2 SICONOS Script for LCS

Script for Problem 2.17

```
#include "SiconosKernel.hpp"
#include <fstream>
#include <string>
using namespace std;

 int main(int argc, char* argv [])
{ // 1
double t0;
double T;            // Total simulation time
double h_step = 5.e-4;// Time step
double xinit_r = 0.0;
double xinit = 1;
```

```
double k = .25;
 string  Modeltitle  = "Problem1";
 try
 {
SP::SiconosVector  init_state_r  (new SiconosVector(1));
       init_state_r  ->setValue(0,  xinit_r );
SP::SiconosVector  init_state  (new SiconosVector(1));
       init_state  ->setValue(0,  xinit );
SP::SiconosVector  control_r (new SiconosVector(1));
SP::SiconosVector  control (new SiconosVector(1));
SP::SimpleMatrix LS_A(new SimpleMatrix(1, 1));
     LS_A->setValue(0 , 0,  -1.0);
SP::SimpleMatrix Int_B(new SimpleMatrix(1,  1));
           Int_B->setValue(0 ,  0,  1.0);
SP::SimpleMatrix Int_C(new SimpleMatrix(1,  1));
           Int_C->setValue(0 ,  0 ,  1.0);
SP::SimpleMatrix Int_D(new SimpleMatrix(1,  1));
           Int_D->setValue(0 ,  0,  .0);
SP::SimpleMatrix LS_A1(new SimpleMatrix(1, 1));
     LS_A1->setValue(0 , 0,  -1.0+k);
SP::SimpleMatrix Int_C1(new SimpleMatrix(1,  1));
           Int_C1->setValue(0 ,  0 ,  1.0+k);

 int  j;
 int  N = 50000;
double  theta  = 0.5;
SimpleMatrix  dataPlot (N+1, 4);
      dataPlot (0,0)  = 0;
      dataPlot (0,1)  =  xinit_r ;
      dataPlot (0,2)  =  xinit ;
        dataPlot (0,3)  = 0;
 for  (j=0;j<N;++j)
 { //Loop time
 t0  = j*h_step;
T = (j+1)*h_step;
 control_r ->setValue(0,  theta*sin(T)+(1-theta)*sin(t0));

 //  First  step:  Compute desired  trajectory
```

```
SP::FirstOrderLinearDS LSCircuitRLCD(new FirstOrderLinearTIDS(init_state_r,
    LS_A, control_r));
SP::FirstOrderLinearTIR LTIRCircuitRLCD(new FirstOrderLinearTIR(Int_C,
    Int_B));
    LTIRCircuitRLCD->setDPtr(Int_D);
    LTIRCircuitRLCD->setePtr(control_r);
SP::NonSmoothLaw NSLaw(new ComplementarityConditionNSL(1));
SP:: Interaction InterCircuitRLCD(new Interaction (NSLaw, LTIRCircuitRLCD));
//
SP::NonSmoothDynamicalSystem CircuitRLCD(new
    NonSmoothDynamicalSystem(t0, T));
// dataPlot(j+1, 0) = CircuitRLCD->t0();
//CircuitRLCD->setTitle(Modeltitle);
CircuitRLCD->insertDynamicalSystem(LSCircuitRLCD);
CircuitRLCD->link(InterCircuitRLCD, LSCircuitRLCD);
InterCircuitRLCD->computeOutput(t0,0);
InterCircuitRLCD->computeInput(t0,0);
//CircuitRLCD->display();

    // -- (1) OneStepIntegrators --
    SP::EulerMoreauOSI OSI_RLCD(new EulerMoreauOSI(theta));

    // -- (2) Time discretisation --
    SP:: TimeDiscretisation TiDiscRLCD(new TimeDiscretisation(t0, h_step));
    // --- (3) one step non smooth problem
    SP::LCP LCP_RLCD(new LCP());

    // -- (4) Simulation setup with (1) (2) (3)
    SP::TimeStepping StratCircuitRLCD(new TimeStepping(CircuitRLCD,
        TiDiscRLCD, OSI_RLCD, LCP_RLCD));
    double h = StratCircuitRLCD->timeStep();
    //boost::timer t;
    //t.restart();
    StratCircuitRLCD->computeOneStep(); //solve!
   // cout<<(*LSCircuitRLCD->x())(0)<<endl;
   (*control)(0) = -k*(theta*(*LSCircuitRLCD->x())(0)
       +(1-theta)*(*init_state_r)(0))+(*control_r)(0);
    init_state_r ->setValue(0, (*LSCircuitRLCD->x())(0));
   dataPlot(j+1, 0) = StratCircuitRLCD->nextTime();
```

```
    dataPlot(j+1, 1) = (*LSCircuitRLCD->x())(0);
// Second step: Compute state x

SP::FirstOrderLinearDS LSCircuitRLCD1(new FirstOrderLinearTIDS(init_state,
    LS_A1, control));
SP::FirstOrderLinearTIR LTIRCircuitRLCD1(new FirstOrderLinearTIR(Int_C1,
    Int_B));
LTIRCircuitRLCD1->setDPtr(Int_D);
LTIRCircuitRLCD1->setePtr(control);
SP::NonSmoothLaw NSLaw1(new ComplementarityConditionNSL(1));
SP::Interaction InterCircuitRLCD1(new Interaction(NSLaw1,
    LTIRCircuitRLCD1));
//

SP::NonSmoothDynamicalSystem CircuitRLCD1(new
    NonSmoothDynamicalSystem(t0, T));
//CircuitRLCD->setTitle(Modeltitle);
CircuitRLCD1->insertDynamicalSystem(LSCircuitRLCD1);
CircuitRLCD1->link(InterCircuitRLCD1, LSCircuitRLCD1);
InterCircuitRLCD1->computeOutput(t0,0);
InterCircuitRLCD1->computeInput(t0,0);
//CircuitRLCD1->display();

    // -- (1) OneStepIntegrators --
    SP::EulerMoreauOSI OSI_RLCD1(new EulerMoreauOSI(theta));

    // -- (2) Time discretisation --
    SP::TimeDiscretisation TiDiscRLCD1(new TimeDiscretisation(t0, h_step));

    // --- (3) one step non smooth problem
    SP::LCP LCP_RLCD1(new LCP());

    // -- (4) Simulation setup with (1) (2) (3)
    SP::TimeStepping StratCircuitRLCD1(new TimeStepping(CircuitRLCD1,
        TiDiscRLCD1, OSI_RLCD1, LCP_RLCD1));
    double h1 = StratCircuitRLCD1->timeStep();
    boost::timer t1;
    t1.restart();
    StratCircuitRLCD1->computeOneStep(); //solve!
```

```
      // cout<<(*LSCircuitRLCD1->x())(0)<<endl;
      init_state ->setValue(0, (*LSCircuitRLCD1->x())(0));
     dataPlot (j+1, 2) = (*LSCircuitRLCD1->x())(0);
     dataPlot (j+1, 3) = (InterCircuitRLCD1->getLambda(0))(0);

} // End Loop time

ioMatrix :: write ("problem1.dat", " ascii ", dataPlot , "noDim");
}
 // --- Exceptions handling ---
   catch (SiconosException e)
   {
     cerr << e. report () << endl;
     return 1;
   }
   catch (...)
   {
     cerr << "Exception caught " << endl;
     return 1;
   }
} // 1
```

Script for Problem 2.21

```
#include "SiconosKernel.hpp"
#include <fstream>
#include <string>
using namespace std;

 int main(int argc, char* argv [])
{ // 1
double t0;
double T;           // Total simulation time
double h_step = 1.e-4;// Time step
double xinit_r = 0.0;
double xinit_r1 = 0.0;
double xinit = 0.5;
double xinit1 = 0.5;
```

```
double k1 = -2.;
double k2 = -1.;
 string  Modeltitle  = "High Dimension Problem";
 try
{
SP::SiconosVector  init_state_r  (new SiconosVector(2));
      init_state_r  ->setValue(0,  xinit_r );
       init_state_r  ->setValue(1,  xinit_r1 );
SP::SiconosVector  init_state  (new SiconosVector(2));
      init_state  ->setValue(0,  xinit );
       init_state  ->setValue(1,  xinit1 );
SP::SiconosVector  control_r (new SiconosVector(2));
SP::SiconosVector  control (new SiconosVector(2));

SP::SimpleMatrix LS_A(new SimpleMatrix(2, 2));
    LS_A->setValue(0 , 0,  -2.0);
    LS_A->setValue(0 , 1,  1.0) ;
    LS_A->setValue(1 , 0,  1.0) ;
    LS_A->setValue(1 , 1,  -2.0);
SP::SimpleMatrix Int_B(new SimpleMatrix(2,  2));
    Int_B->setValue(0 ,  0,  1.0) ;
    Int_B->setValue(0 ,  1,  1.0) ;
    Int_B->setValue(1 ,  0,  0.0) ;
    Int_B->setValue(1 ,  1,  -1.0);
SP::SimpleMatrix Int_C(new SimpleMatrix(2,  2));
    Int_C->setValue(0 ,  0,  1.0) ;
    Int_C->setValue(1 ,  0,  1.0) ;
    Int_C->setValue(0 ,  1,  0.0) ;
    Int_C->setValue(1 ,  1,  -1.0);
SP::SimpleMatrix Int_D(new SimpleMatrix(2,  2));
SP::SimpleMatrix LS_A1(new SimpleMatrix(2, 2));
    LS_A1->setValue(0 , 0,  -2.0);
    LS_A1->setValue(0 , 1,  1.0) ;
    LS_A1->setValue(1 , 0,  -1.);
    LS_A1->setValue(1 , 1,  -3.);

 int  j ;
 int  N = 100000;
double  theta  = .5;
```

```
SimpleMatrix dataPlot (N+1, 7);
    dataPlot (0,0) = 0;
    dataPlot (0,1) = xinit_r ;
dataPlot (0,2) = xinit_r1 ;
    dataPlot (0,3) = xinit ;
dataPlot (0,4) = xinit1 ;
        dataPlot (0,5) = 0;
dataPlot (0,6) = 0;
for (j=0;j<N;++j)
{//2
t0 = j*h_step;
T = (j+1)*h_step;
control_r ->setValue(0, 0);
control_r ->setValue(1, theta*sin(T)+(1-theta)*sin(t0));

// First round
SP::FirstOrderLinearDS LSCircuitRLCD(new FirstOrderLinearTIDS(init_state_r ,
    LS_A,control_r));
SP::FirstOrderLinearTIR LTIRCircuitRLCD(new FirstOrderLinearTIR(Int_C,
    Int_B));
LTIRCircuitRLCD->setDPtr(Int_D);
//LTIRCircuitRLCD->setePtr(control_r);
SP::NonSmoothLaw NSLaw(new ComplementarityConditionNSL(2));
SP:: Interaction InterCircuitRLCD(new Interaction (NSLaw, LTIRCircuitRLCD));
//
SP::NonSmoothDynamicalSystem CircuitRLCD(new
    NonSmoothDynamicalSystem(t0, T));
// dataPlot(j+1, 0) = CircuitRLCD->t0();
//CircuitRLCD->setTitle(Modeltitle);
CircuitRLCD->insertDynamicalSystem(LSCircuitRLCD);
CircuitRLCD->link(InterCircuitRLCD, LSCircuitRLCD);
InterCircuitRLCD->computeOutput(t0,0);
InterCircuitRLCD->computeInput(t0,0);
//CircuitRLCD->display();
    // -- (1) OneStepIntegrators --
    SP::EulerMoreauOSI OSI_RLCD(new EulerMoreauOSI(theta));

    // -- (2) Time discretisation --
    SP:: TimeDiscretisation TiDiscRLCD(new TimeDiscretisation(t0, h_step));
```

```
    // --- (3) one step non smooth problem
    SP::LCP LCP_RLCD(new LCP());

    // -- (4) Simulation setup with (1) (2) (3)
    SP::TimeStepping StratCircuitRLCD(new TimeStepping(CircuitRLCD, TiDiscRLCD, OSI_RLCD, LCP_RLCD));
    double h = StratCircuitRLCD->timeStep();
    boost :: timer t;
    t. restart ();
    StratCircuitRLCD->computeOneStep(); //solve!
   // cout<<(*LSCircuitRLCD->x())(0)<<endl;
   (* control ) (0) = 0;
   (* control ) (1) = -theta*(k1*(*LSCircuitRLCD->x())(0)+k2*(*LSCircuitRLCD->x())(1))
   -(1-theta)*(k1*(* init_state_r )(0)+k2*(* init_state_r )(1))+(* control_r )(1);
    init_state_r ->setValue(0, (*LSCircuitRLCD->x())(0));
    init_state_r ->setValue(1, (*LSCircuitRLCD->x())(1));
   dataPlot (j+1, 0) = StratCircuitRLCD->nextTime();
   dataPlot (j+1, 1) = (*LSCircuitRLCD->x())(0);
   dataPlot (j+1, 2) = (*LSCircuitRLCD->x())(1);
// Second round
SP::FirstOrderLinearDS LSCircuitRLCD1(new FirstOrderLinearTIDS(init_state , LS_A1, control));
SP:: FirstOrderLinearTIR LTIRCircuitRLCD1(new FirstOrderLinearTIR(Int_C, Int_B));
LTIRCircuitRLCD1->setDPtr(Int_D);
//LTIRCircuitRLCD1->setePtr(control);
SP::NonSmoothLaw NSLaw1(new ComplementarityConditionNSL(2));
SP:: Interaction InterCircuitRLCD1(new Interaction (NSLaw1, LTIRCircuitRLCD1));
//
SP::NonSmoothDynamicalSystem CircuitRLCD1(new NonSmoothDynamicalSystem(t0, T));
//CircuitRLCD->setTitle(Modeltitle);
CircuitRLCD1->insertDynamicalSystem(LSCircuitRLCD1);
CircuitRLCD1->link(InterCircuitRLCD1, LSCircuitRLCD1);
InterCircuitRLCD1->computeOutput(t0,0);
InterCircuitRLCD1->computeInput(t0,0);
//CircuitRLCD1->display();
```

```
    // -- (1) OneStepIntegrators --
    SP::EulerMoreauOSI OSI_RLCD1(new EulerMoreauOSI(theta));

    // -- (2) Time discretisation --
    SP:: TimeDiscretisation TiDiscRLCD1(new TimeDiscretisation(t0, h_step));
    // --- (3) one step non smooth problem
    SP::LCP LCP_RLCD1(new LCP());

    // -- (4) Simulation setup with (1) (2) (3)
    SP::TimeStepping StratCircuitRLCD1(new TimeStepping(CircuitRLCD1,
        TiDiscRLCD1, OSI_RLCD1, LCP_RLCD1));
    double h1 = StratCircuitRLCD1->timeStep();
    boost :: timer t1 ;
    t1 . restart () ;
    StratCircuitRLCD1->computeOneStep(); //solve!
    // cout<<(*LSCircuitRLCD1->x())(0)<<endl;
    init_state ->setValue(0, (*LSCircuitRLCD1->x())(0));
 init_state ->setValue(1, (*LSCircuitRLCD1->x())(1));
   dataPlot (j+1, 3) = (*LSCircuitRLCD1->x())(0);
dataPlot (j+1, 4) = (*LSCircuitRLCD1->x())(1);
   dataPlot (j+1, 5) = (InterCircuitRLCD1->getLambda(0))(0);
dataPlot (j+1, 6) = (InterCircuitRLCD1->getLambda(0))(1);
} // 2
cout<<"Completed!"<<endl;
ioMatrix :: write ("High_Dimension_Problem.dat", " ascii ", dataPlot , "noDim");
}
 // --- Exceptions handling ---
  catch (SiconosException e)
  {
    cerr << e. report () << endl;
    return 1;
  }
  catch (...)
  {
    cerr << "Exception caught " << endl;
    return 1;
  }
} // 1
```

B.3 YALMIP codes for solving (Q_κ) and (K)

Assuming that the matrices $A, B, C, D, E,$ and F already exist in the environment.

Optimal ε

```
k=1;
eps_0=1;
n=size(A,1);
m=size(E,2);
Q=sdpvar(n);
L=sdpvar(m,n,'full');
tol = 1e-6;

% Step 1: Find a lower bound
cons=[Q>=eye(n),Q*A'+A*Q+L'*E'+E*L+eps_0*Q<=0,
    B-Q*C'-L'*F'==0,[k*eye(m), L;L', k*eye(n)]>=0];
ops = sdpsettings('solver','sedumi','verbose',0,'warning',0);
sol = optimize(cons,[], ops);
while sol.problem==1 %F_eps_0 is empty
    eps_0 = eps_0/2;
    cons=[Q>=eye(n),Q*A'+A*Q+L'*E'+E*L+eps_0*Q<=0,
        B-Q*C'-L'*F'==0,[k*eye(m), L;L', k*eye(n)]>=0];
    sol = optimize(cons,[], ops);
end
eps_lower=eps_0;

% Step 2: Find an upper bound
cons=[Q>=eye(n),Q*A'+A*Q+L'*E'+E*L+eps_0*Q<=0,
    B-Q*C'-L'*F'==0,[k*eye(m), L;L', k*eye(n)]>=0];
sol = optimize(cons,[], ops);
while ~(sol.problem==1) %F_eps_0 is nonempty
    eps_0 = eps_0*2;
    cons=[Q>=eye(n),Q*A'+A*Q+L'*E'+E*L+eps_0*Q<=0,
        B-Q*C'-L'*F'==0,[k*eye(m), L;L', k*eye(n)]>=0];
    sol = optimize(cons,[], ops);
end
eps_upper=eps_0;

```

```
% Step 3:  Start  bisection
eps_works = eps_lower;
while  (eps_upper−eps_lower)>tol
      eps_test  = (eps_upper+eps_lower)/2;
     cons=[Q>=eye(n),Q∗A'+A∗Q+L'∗E'+E∗L+eps_test∗Q<=0,
          B−Q∗C'−L'∗F'==0,[k∗eye(m), L;L', k∗eye(n)]>=0];
     sol  = optimize(cons ,[], ops);
  if  sol .problem==1
     eps_upper = eps_test ;
  else
     eps_lower = eps_test ;
     eps_works = eps_test ;
     Qworks = double(Q);
     Lworks = double(L);
  end
end

% Recover gain  K
K=Lworks∗Qworks^(−1)
```

Optimal κ

```
eps_max=4;
n=size(A,1);
m=size(E,2);
sdpvar  k;
Q=sdpvar(n);
L=sdpvar(m,n,' full ');
cons=[Q>=eye(n),k>=0, Q∗A'+A∗Q+L'∗E'+E∗L+eps_max∗Q<=0,
     B−Q∗C'−L'∗F'==0,[k∗eye(m), L;L', k∗eye(n)]>=0];
ops = sdpsettings (' solver ' ,'sedumi','verbose' ,0, 'warning',0) ;
sol  = optimize(cons,k,ops);
k = value(k);
Q = value(Q);
L = value(L);
K = L∗Q^(−1);
```

Bibliography

[1] V. Acary and B. Brogliato. *Numerical Methods for Nonsmooth Dynamical Systems: Applications in Mechanics and Electronics*, volume 35. Springer Verlag, 2008.

[2] S. Adly. *A Variational Approach to Nonsmooth Dynamics: Applications in Unilateral Mechanics and Electronics*. Springer, 2017.

[3] B. Brogliato and D. Goeleven. Well-posedness, stability and invariance results for a class of multivalued Lur'e dynamical systems. *Nonlinear Analysis, Theory, Methods, and Applications*, 74:195–212, 2011.

[4] B. Brogliato, R. Lozano, B. Maschke, and O. Egeland. *Dissipative Systems Analysis and Control, Theory and Applications*. Springer Verlag, London, U.K., 2nd edition, 2007.

[5] B. Brogliato and L. Thibault. Existence and uniqueness of solutions for non-autonomous complementarity dynamical systems. *Journal of Convex Analysis*, 17(3-4):961–990, 2010.

[6] M.K. Camlibel, W.P.M.H. Heemels, and J.M. Schumacher. On linear passive complementarity systems. *European Journal of Control*, 8(3):220–237, 2002.

[7] P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali. *LMI Control Toolbox For Use with MATLAB*. The MathWorks, Inc., 1st edition.

[8] D. Goeleven, D. Montreanu, and V.V. Montreanu. On the stability of stationary solutions of first order evolution variational inequalities. *Advances in Nonlinear Variational inequalities*, 6:1–30, 2013.

[9] W.P.M.H. Heemels and B. Brogliato. The complementarity class of hybrid dynamical systems. *European Journal of Control*, 9(2-3):322–360, 2003.

[10] K.G. Murty. *Linear Complementarity, Linear and Nonlinear Programming*. Heldermann, 1988.